

**UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR**



Doble Grado en Ingeniería Informática y Matemáticas

TRABAJO FIN DE GRADO

**La Realidad Aumentada en el entorno de la
educación especial**

**Autor: Alejandro López Sanz
Tutor: Javier Gómez Escribano
Ponente: Germán Montoro Manrique**

junio 2019

Algunos derechos reservados.

Este trabajo está bajo licencia Creative Commons

<http://creativecommons.org/licenses/by-nc-sa/3.0/>

Esta obra se puede copiar, distribuir y comunicar públicamente la obra así como crear obras derivadas bajo las siguientes condiciones:

- Debe reconocer los créditos manteniendo la autoría original y añadiendo la autoría de las modificaciones indicando de forma expresa y bien visible que el autor original no manifiesta ningún tipo de apoyo a las modificaciones realizadas así como al uso que se da de esta obra.
- No se puede utilizar esta obra con fines comerciales.
- Las modificaciones o ediciones de esta obra deben compartirse bajo una licencia idéntica a esta.

La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual (*arts. 270 y sgts. del Código Penal*).

DERECHOS RESERVADOS

© 3 de Noviembre de 2017 por UNIVERSIDAD AUTÓNOMA DE MADRID

Francisco Tomás y Valiente, nº 1

Madrid, 28049

Spain

Alejandro López Sanz

La Realidad Aumentada en el entorno de la educación especial

Alejandro López Sanz

Madrid, España

IMPRESO EN ESPAÑA – PRINTED IN SPAIN

A mi familia y mis amigos

*Los ordenadores son inútiles,
sólo pueden darnos respuestas.*

Pablo Picasso

AGRADECIMIENTOS

A todas las personas que han formado parte de esta maravillosa etapa de cinco años de mi vida.

En primer lugar, a mi tutor Javier; por su paciencia y estar siempre disponible para ayudarme con este trabajo.

Y, por supuesto, a mis amigos y mi familia; en especial a mis padres y mi hermano, porque sin su apoyo incondicional nada de esto habría sido posible.

A todos, muchas gracias.

RESUMEN

El gran crecimiento que están experimentando las nuevas tecnologías en el presente siglo está provocando importantes transformaciones en la inmensa mayoría de los aspectos que conforman nuestra vida diaria. La tecnología es un pilar cada vez más fundamental de las sociedades actuales; como también lo es la educación, y por ello es indispensable poder incorporar estas nuevas tecnologías existentes en el ámbito del sector educativo. De la enciclopedia de nuestros padres se pasó a la Encarta y después a la Wikipedia; y ahora el bolígrafo se ha sustituido por las pantallas táctiles, y los dispositivos electrónicos han pasado a ser un elemento de ayuda para los estudiantes.

La realización del proyecto busca conseguir una aplicación final para Android, partiendo de un trabajo previo realizado a partir del cual se comenzará a desarrollar, que se ajuste a las necesidades relativas con el proceso de aprendizaje en términos de las operaciones aritméticas y la autoevaluación de dichos resultados por parte de los estudiantes, de manera que la aplicación sea útil para dichos procesos y pueda ser empleada en el contexto de la educación especial. Es por ello que la aplicación, a medida que sea desarrollada; deberá ser validada por profesionales de la educación especial.

El sistema será utilizado tanto por profesores como por alumnos. Permite a los docentes la gestión de alumnos, la creación de operaciones aritméticas y la asignación de dichos ejercicios a los alumnos. El profesor imprime estas fichas y se las entrega a sus alumnos; que una vez las realicen, podrán determinar a través de la aplicación si ese resultado es o no correcto. Para ello, se hará uso de la tecnología de la Realidad Aumentada; y la aplicación almacenará una copia de dicha ficha solucionada con el resultado de la autoevaluación.

A lo largo de este documento se detalla todo el proceso realizado durante dicho proyecto, partiendo de una introducción que permita contextualizar el ámbito en el que nos encontramos, prosiguiendo con las tareas de análisis, diseño y desarrollo; para finalmente determinar el punto en el que se encuentra la versión final de la aplicación, sus posibles líneas futuras de desarrollo y las conclusiones extraídas a partir del trabajo efectuado.

PALABRAS CLAVE

Aplicación, Android, Educación, Realidad Aumentada, QR, Autoevaluación, Gestión Docente

ABSTRACT

The great growth that new technologies are suffering in this century is causing important changes in most of the aspects that are part of our daily life. Technology is increasingly important in today's societies; as is education, and therefore it is essential to be able to incorporate the new existing technologies in the education sector. From the encyclopedia, we have move on to Encarta and then to Wikipedia; and nowadays the pen is being replaced by touch screens, and electronic devices have become an aid element for the students.

This software project seeks to achieve an application for Android, starting from a previous work which will be the start point to begin the development. This project will adjust to the relative needs with the learning process in terms of the arithmetic operations and the self-evaluation of the results by the students. The application is useful for these processes, and it can be used in the context of special education. Thus, during the development process, the application must be checked by the teachers.

The software will be used by both teachers and students. It allows teachers to manage students, create arithmetic operations and assign these exercises to students. The teacher can prints these exercises and give them to their students. When the student does that exercise, he can determine in the application whether the result is correct or not. To do this, the Augmented Reality technology will be used; and the application must save a copy of that submitted exercise that also includes the result of self-evaluation.

The document details the entire process associated with this project, starting with an introduction that allows us to contextualize the scope in which we are; and then treating tasks related to analysis, design and development. Finally, the point where the final version of the application is, its possible future works and the conclusions of the work done are determined.

KEYWORDS

Application, Android, Education, Augmented Reality, QR, Self-Evaluation, Teaching Management

ÍNDICE

1	Introducción	1
1.1	Motivación	1
1.2	Objetivos	2
1.3	Organización de la memoria	3
2	Estado del Arte	5
2.1	Trabajos previos	5
2.1.1	FETCH! Lunch Rush	5
2.1.2	AR Flashcards Addition	6
2.1.3	Math Alive	8
2.2	Conclusiones extraídas	8
3	Definición del Proyecto	11
3.1	Estado Inicial de la Aplicación	11
3.2	Estructura Lógica	13
3.3	Análisis de Requisitos	13
3.3.1	Requisitos Funcionales	14
3.3.2	Requisitos No Funcionales	17
3.4	Diseño de la Interacción del Sistema	18
3.5	Análisis y Propuesta de Solución	23
4	Desarrollo	25
4.1	Herramientas empleadas	25
4.1.1	Lenguajes de programación	25
4.1.2	Librerías	25
4.1.3	Dispositivos	28
4.2	Arquitectura Software	28
4.2.1	Arquitectura de realidad aumentada y captura de autoevaluación	29
4.2.2	Arquitectura de ejercicios	30
4.2.3	Arquitectura de alumnos	30
4.2.4	Arquitectura de asignaciones	31
4.2.5	Arquitectura de impresión de fichas de ejercicios	31
4.2.6	Arquitectura de la generación automática de ejercicios	32
4.2.7	Arquitectura de la interfaz de usuario	33

4.3 Modelo de Datos	34
5 Integración, pruebas y resultados	37
6 Conclusiones y trabajo futuro	39
6.1 Conclusiones	39
6.2 Trabajo Futuro	40
Bibliografía	41
Apéndices	43
A Manual de Usuario para Alumnos	45
B Manual de Usuario para Docentes	49
C Capturas del Estado Inicial	57
D Capturas del Estado Final	63
E Ejemplo de ficha de ejercicios	81

LISTAS

Lista de figuras

2.1	Marcador de Realidad Aumentada de FETCH! Lunch Rush	6
2.2	Captura de pantalla de FETCH! Lunch Rush	6
2.3	Marcador de Realidad Aumentada de AR Flashcards Addition	7
2.4	Capturas de pantalla de AR Flashcards Addition	7
2.5	Captura de pantalla de Math Alive	8
3.1	Subsistemas de la aplicación MathematicAR	13
4.1	Marcador de Realidad Aumentada de MathematicAR	26
4.2	Captura de pantalla de un Floating Action Button Speed Dial	27
4.3	Posibles interacciones entre las diferentes vistas de la aplicación	28
4.4	Arquitectura de los componentes asociados con la Realidad Aumentada	29
4.5	Arquitectura de la gestión de los ejercicios	30
4.6	Arquitectura de la gestión de los alumnos	31
4.7	Arquitectura de la gestión de las asignaciones	31
4.8	Arquitectura para la impresión de fichas	32
4.9	Diagrama de las clases implicadas en la interfaz	33
4.10	Modelo Entidad-Relación de MathematicAR	34
A.1	Menú principal de MathematicAR	45
A.2	Ficha enfocada con la cámara	46
A.3	Mensaje de entrega satisfactoria	47
B.1	Menú principal de MathematicAR	49
B.2	Lista con los ejercicios existentes	50
B.3	Posibles maneras de generar operaciones	51
B.4	Opciones disponibles sobre uno o varios ejercicios seleccionados	51
B.5	Vista de las asignaciones	52
B.6	Posibles visiones de una entrega concreta	53
B.7	Lista con los alumnos existentes	54
B.8	Diálogo mostrado al crear un nuevo alumno	54
B.9	Opciones disponibles sobre uno o varios alumnos seleccionados	55
C.1	Menú principal de la aplicación	57

C.2	Vista inicial de Corrige tus Ejercicios	57
C.3	Resultado de la operación de la ficha de ejercicios	58
C.4	Lista con los ejercicios existentes	58
C.5	Menú con el tipo de operación a crear	59
C.6	Diálogo para la creación de la operación	59
C.7	Selección de ejercicios	60
C.8	Ejemplo de ficha de ejercicios	60
C.9	Vista de los ajustes disponibles	61
C.10	Diálogo con las resoluciones de cámara disponibles	61
D.1	Vista inicial previa a la carga del menú principal	63
D.2	Menú principal de MathematicAR	63
D.3	Vista inicial de Corrige tus Ejercicios	64
D.4	Diálogo relativo a salir al menú principal	64
D.5	Mensaje de ayuda	65
D.6	Transición cuando la cámara capta el marcador y lee el código QR	65
D.7	Posibles mensajes mostrados al realizar una entrega	66
D.8	Lista con los ejercicios existentes	66
D.9	Posibles estados de la opción de búsquedas	67
D.10	Acciones disponibles para la creación de ejercicios	68
D.11	Diálogo del tipo de operación a crear	68
D.12	Diálogo para la creación de un conjunto de operaciones	69
D.13	Diálogo para la creación de una operación	69
D.14	Proceso de colapso y expansión de los elementos de la interfaz	70
D.15	Selección de elementos de la interfaz	71
D.16	Opciones disponibles sobre uno o varios elementos seleccionados	71
D.17	Impresión de un ejercicio seleccionado	72
D.18	Impresión de un ejercicio de un alumno seleccionado	72
D.19	Proceso de borrado mediante <i>swipe</i>	73
D.20	Mensaje de eliminación de un elemento	73
D.21	Lista con las asignaciones existentes	74
D.22	Posibles estados de la opción de búsquedas	74
D.23	Posibles estados de la opción de búsquedas exitosas	75
D.24	Interfaz tras expandir elementos	76
D.25	Posibles visiones de una entrega concreta	76
D.26	Lista con los alumnos existentes	77
D.27	Posibles estados de la opción de búsquedas	78
D.28	Diálogo para la creación de alumnos	79
D.29	Selección de alumnos	79

D.30 Opciones disponibles sobre uno o varios alumnos seleccionados	79
D.31 Vista de los ajustes disponibles	80
D.32 Diálogo con las resoluciones de cámara disponibles.....	80
E.1 Ejemplo de ficha de ejercicios de MathematicAR.....	81

INTRODUCCIÓN

El presente documento constituye un informe en el que se detallará el proyecto software realizado en colaboración con el Colegio de Educación Especial Alenta, participando en el desarrollo de una aplicación para dispositivos Android; orientada a la asignatura de Matemáticas y haciendo uso de la Realidad Aumentada, que permita realizar operaciones matemáticas básicas y que estas sean auto-evaluadas por los alumnos.

1.1. Motivación

La educación es un pilar básico para la formación de las personas que conformarán la sociedad futura. Con el paso del tiempo, especialmente en los últimos años, la aparición y el crecimiento exponencial de las nuevas tecnologías ha propiciado que éstas se hayan incorporado poco a poco a todos los ámbitos de nuestra vida, incluida la educación. Es por ello que el uso de la tecnología en el ámbito educativo está cada vez más extendido.

De acuerdo con un estudio elaborado por la empresa Samsung en el año 2016 [1] a profesores españoles en relación con las nuevas tecnologías en las aulas, en torno al 80 % considera que incrementa la autonomía de los alumnos en su propio aprendizaje, colaboran entre ellos más y las clases son más entretenidas, mostrando un mayor esfuerzo por aprender. Además, tres de cada cuatro profesores (74 %) asegura que se comprenden mejor los contenidos impartidos.

Los métodos de aprendizaje son cambiantes con el paso del tiempo; a medida que pasan los años la manera de enseñar es muy diferente: no se nos ha enseñado de la misma manera que a nuestros padres, y muy probablemente a nuestros hijos se les enseñará de manera diferente a nosotros.

Por tanto, es necesario llevar a cabo herramientas que permitan a los docentes sumarse a la revolución digital e incorporarla a las aulas, con el objetivo de facilitar la transmisión de los contenidos a sus alumnos y despertar la curiosidad de los mismos.

En términos de la educación especial, la existencia de este tipo de herramientas se hace aún más necesaria, pues acentúa todavía más esta facilidad de transmisión de contenidos. Además, en este

contexto, es importante el análisis individual de cada alumno, y conocer las necesidades de aprendizaje de los mismos.

La incorporación de las nuevas tecnologías permite por tanto al equipo docente elaborar contenidos personalizados, los cuales posteriormente serán evaluados, y les permitirán adaptar los contenidos a los diferentes ritmos de aprendizaje existentes en el aula.

Esta incorporación propicia además el crecimiento en la autonomía en su propio aprendizaje, como se ha indicado anteriormente. Es aquí donde el concepto de la autoevaluación cobra especial relevancia. La autoevaluación es una metodología que no reemplaza a otros métodos de evaluación, sino que refuerza estos cuando son empleados de manera conjunta. Luego no se limita únicamente al alumando; sino que es responsabilidad del docente evaluar tanto las tareas realizadas por el alumno como la autoevaluación de la misma.

1.2. Objetivos

En la presente sección se van a exponer los principales objetivos perseguidos en el proyecto, así como delimitar el alcance de la aplicación que va a ser desarrollada.

El objetivo principal de este trabajo es llevar a cabo el desarrollo de una herramienta software Android basada en técnicas de realidad aumentada; y dirigida hacia el sector de la educación, más concretamente a la educación especial; partiendo de un proyecto existente [2] sobre el cual se añadirán las funcionalidades requeridas, y realizando las mejoras y correcciones de errores que se encuentren en dicho proyecto durante el desarrollo del mismo.

Dicha herramienta software sirve como apoyo al proceso de aprendizaje de operaciones aritméticas básicas (suma, resta, multiplicación y división) llevado a cabo por parte del equipo docente del Colegio de Educación Especial Alenta.

Además, incorporará funcionalidades relativas al proceso de autoevaluación por parte de los alumnos de las diferentes fichas de ejercicios que el profesor les asigne; permitiendo al alumnado determinar si el resultado de la operación aritmética realizada es correcto o no.

Permitirá al equipo docente crear los ejercicios, crear a los alumnos y asignar dichos ejercicios a los alumnos que corresponda; pudiendo posteriormente imprimir las fichas con los ejercicios de cada uno de los alumnos. Una vez que el alumno ha efectuado la entrega de uno de los ejercicios que el profesor le ha entregado; el docente es capaz de visualizar la entrega realizada, incluyendo la autoevaluación de la misma.

De esta manera, el profesor es capaz de evaluar tanto el proceso de autoevaluación como el resultado de la operación llevada a cabo por parte de su alumno. Así, el docente podrá ser capaz de establecer con exactitud en qué estado del proceso de aprendizaje se encuentra cada uno de sus

alumnos; pudiendo adaptar los ejercicios asignados en base a dicho proceso. Con el desarrollo de este proyecto se pretende también otorgar importancia a este proceso de aprendizaje; y, para facilitar esta tarea a los docentes, permitirles generar conjuntos de ejercicios de un determinado tipo de acuerdo con ese proceso.

Por último, con el fin de que la aplicación se ajuste a lo esperado por parte del equipo docente del Colegio de Educación Especial Alenta, se mantendrá un contacto lo más constante posible de forma que se recoja el *feedback* ofrecido por los mismos, aplicando si procede lo indicado al proyecto.

1.3. Organización de la memoria

Este documento se encuentra estructurado en seis capítulos, el apartado de la Bibliografía; y cuenta además con un total de cinco Apéndices. A continuación, se describe el contenido de cada uno de ellos.

El primer capítulo sirve de presentación de la herramienta software que se va a desarrollar; mostrando una visión general de ésta. Se especifican la motivación de dicho software, así como los objetivos que se persiguen con el mismo.

En segundo lugar, el capítulo tratado aborda el estado del arte; centrándose en el estudio de las diferentes aplicaciones y tecnologías existentes en la actualidad en relación con el contexto del proyecto centrándonos, esencialmente, en la Realidad Aumentada. Finalmente, se extraerán las pertinentes conclusiones del análisis efectuado, indicando qué funcionalidades son interesantes para nuestro proyecto y cuáles no.

En tercer lugar, el capítulo tratado en el informe del proyecto será la definición del mismo, en el cual primeramente se especificará el estado en el que se encuentra el proyecto previo al comienzo del desarrollo, y a continuación se explicará la organización lógica del sistema, con los subsistemas que lo conforman; además de llevar a cabo la especificación del conjunto inicial de requisitos funcionales y no funcionales. Finalmente, nos enfocaremos en el diseño conceptual de la iteración por parte de los usuarios con el sistema, el análisis de las soluciones existentes para satisfacer dichos requisitos y las soluciones elegidas.

El cuarto capítulo especifica primeramente el conjunto de las herramientas empleadas durante el proceso de desarrollo del proyecto, centrándose especialmente en las librerías de mayor importancia. Posteriormente, se describe la arquitectura de los diferentes componentes que forman parte de la aplicación; y finalmente se explica detalladamente el modelo de datos de esta.

En el quinto capítulo se presenta el trabajo relativo a la integración de los diferentes módulos que se van a añadir al proyecto, las diferentes pruebas (tanto unitarias como de integración) efectuadas sobre los mismos; además los resultados obtenidos y problemas encontrados en relación con dichas pruebas.

En el último de los seis capítulos, se precisarán las conclusiones extraídas a partir del trabajo, indicando también posibles extensiones del mismo a llevar a cabo en un futuro.

Seguidamente, se presenta un apartado con las referencias utilizadas durante todo el proceso de desarrollo; y finalmente se tienen los Apéndices.

Los manuales de usuario, tanto para los alumnos como para el personal docente se encuentran en el Apéndice A y Apéndice B, respectivamente. En el Apéndice C se adjuntan las capturas correspondientes al estado en el que se encontraba la aplicación al comienzo del desarrollo; mientras que en el Apéndice D también se disponen capturas de pantalla de la aplicación, pero en este caso correspondientes al estado final de la misma. Finalmente, en el Apéndice E se muestra una ficha de ejercicios de nuestra aplicación, denominada MathematicAR, de ejemplo.

ESTADO DEL ARTE

Este capítulo se centrará en el análisis de las propuestas existentes en relación con los dos focos principales en torno a los que girará el posterior desarrollo del proyecto: la Realidad Aumentada orientada a la educación y la gestión de tareas entregadas y autoevaluación; para posteriormente presentar las conclusiones extraídas de dicho análisis.

2.1. Trabajos previos

El proceso de investigación de los trabajos realizados previamente permitirá establecer una visión crítica acerca de estos. De esta forma, una vez elaborada dicha investigación las conclusiones obtenidas serán empleadas durante todo el proceso de desarrollo del proyecto.

Las soluciones existentes en el ámbito de la Realidad Aumentada orientada a la educación son muy abundantes en la actualidad. Sin embargo, dicha cantidad se reduce sustancialmente cuando nos centramos en la educación especial, incluso sin tener en cuenta requerimientos asociados con la Realidad Aumentada; caso en los que es incluso inferior.

Por ello, se ha determinado centrar el estudio de dichas soluciones enfocándolas a la rama de las Matemáticas, pues se considera que de ellas se pueden extraer conclusiones más relevantes para el proyecto que de un número tan reducido; pudiendo adaptarlas si procede a la educación especial.

2.1.1. FETCH! Lunch Rush

FETCH! Lunch Rush [3] es una aplicación móvil gratuita disponible únicamente para dispositivos que cuenten con iOS. El funcionamiento de la misma se basa en la creación de un número determinado de jugadores (entre uno y cuatro) diferentes, cada uno de los cuales debe resolver diferentes operaciones matemáticas. El sistema otorga una puntuación a cada uno de los jugadores en base al acierto o error en la operación realizada, y también el tiempo requerido para realizarla.

Para poder realizar la operación se hace uso de la Realidad Aumentada: se tiene un conjunto

de cartas o fichas, cada una de las cuales está formada por el marcador de la Realidad Aumentada diferente en cada caso y un número; generando una ficha como la de la figura 2.1 numerada, entre uno y diez.

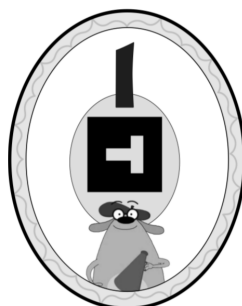


Figura 2.1: Marcador de Realidad Aumentada de FETCH! Lunch Rush

En el turno de cada jugador, se muestra en la parte superior de la pantalla la operación; y se debe enfocar a la ficha cuyo número sea la solución a la misma. Posteriormente, una vez confirmada la selección del resultado, se indica si es correcta o no; continuando a la siguiente operación o al final del turno si era la última operación de este. Cuando se detecta uno de los marcadores de la Realidad Aumentada, se muestran tantas piezas de sushi como el número de dicha carta enfocada con la cámara; tal como se observa en la figura 2.2.

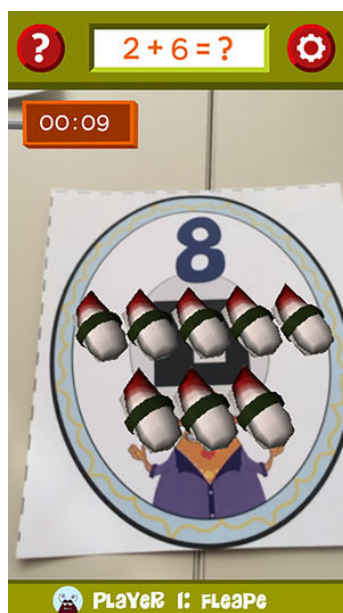


Figura 2.2: Captura de pantalla de FETCH! Lunch Rush

2.1.2. AR Flashcards Addition

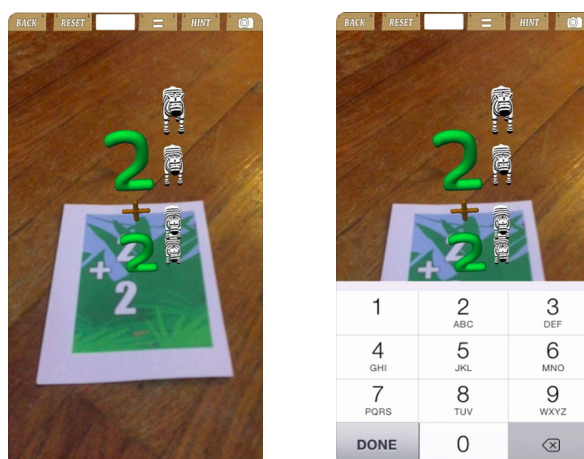
AR Flashcards Addition [4] se encuentra disponible, al igual que en el caso anterior, únicamente para dispositivos que poseen iOS. Sin embargo, es una aplicación de pago. El funcionamiento de esta

aplicación está restringido a las operaciones de suma, y consta de dos vistas principales: una que permite hacer test de sumas y la correspondiente con la Realidad Aumentada, que es la que se va a analizar más detalladamente. El funcionamiento es muy similar a la aplicación anterior: se emplea la cámara del dispositivo en busca del marcador, que en este caso es el mostrado en la figura 2.3. Estas fichas se encuentran accesibles en la red.



Figura 2.3: Marcador de Realidad Aumentada de AR Flashcards Addition

Cuando dicho marcador es detectado, se muestra la operación en cuestión y, para facilitar la resolución, el número de animales totales (correspondiente a la solución de la operación) como muestra la figura 2.4(a). La parte superior de la vista posee varias opciones, entre las que destaca un campo en el que el usuario introduce el resultado de la operación; que es verificado pulsando el botón que se encuentra a su derecha; tal como se observa en la figura 2.4(b). Además, permite almacenar una imagen en el dispositivo de lo que está siendo capturado por la cámara.



(a) Se muestra la operación

(b) Se introduce el resultado

Figura 2.4: Capturas de pantalla de AR Flashcards Addition

2.1.3. Math Alive

Math Alive [5] se define como un kit de matemáticas complementario que emplea la Realidad Aumentada y mecánicas de juegos para el proceso de enseñanza de las matemáticas. No es una aplicación móvil, si no un programa para ordenadores con sistema operativo Windows o macOS. Debido a que es necesario disponer del kit de esta para poder emplearla, para analizar la misma se va a hacer uso de los numerosos vídeos disponibles en la red acerca del funcionamiento de la misma.

Uno de los juegos que posee se basa en completar correctamente un patrón de números y colores concreto, como se muestra en la figura 2.5. Cuando la cámara, en este caso del ordenador, detecta el marcador de la ficha de un determinado número, hace uso de la Realidad Aumentada para mostrarlo sobre dicha ficha. Además, es necesario no sólo determinar el número del patrón sino también el color del mismo. Una vez situadas tanto el número como el color; el usuario puede presionar el botón de la parte inferior obteniendo instantáneamente la resolución de dicho ejercicio.



Figura 2.5: Captura de pantalla de Math Alive

2.2. Conclusiones extraídas

La sección anterior nos ha mostrado varios ejemplos de sistemas existentes en términos de la Realidad Aumentada para la educación asociada con las Matemáticas, más concretamente con las operaciones aritméticas; pero también en algunos de los casos, relacionados con procesos asociados con la evaluación del resultado por parte de dichos sistemas.

El análisis realizado supone un punto de partida en el desarrollo del proyecto, puesto que ahora ya se dispone de un mayor conocimiento acerca de las diferentes posibilidades ya existentes; pudiendo

elaborar un enfoque que se adecúe a los objetivos descritos en la sección 1.2 del capítulo previo.

Cabe destacar que, evidentemente, son muy numerosos los sistemas existentes. Muchos de ellos han sido probados, pero se ha decidido incorporar a este trabajo únicamente aquellos que se han considerado relevantes para el mismo; intentando también no añadir varios sistemas con características a nivel de funcionalidad muy similares.

En cuanto a las conclusiones a extraer de estos trabajos, en primer lugar destacar que, aunque las aplicaciones de Realidad Aumentada en el sector educativo son numerosas; si nos restringimos a la rama de las Matemáticas el número se reduce considerablemente.

Establecer unas limitaciones o penalizaciones en relación con el tiempo que necesita el alumno para realizar una operación no es una funcionalidad deseable, puesto que lo que se busca es que la operación sea realizada correctamente; no que se realice rápido. Por otro lado, el hecho de que el usuario mediante una interacción con la aplicación establezca el momento en el que desea confirmar el resultado es una característica deseable.

En el caso de nuestra aplicación, la acción de introducir en un campo disponible en la interfaz de la misma el resultado de la operación no se va a realizar puesto que la operación va a ser entregada en papel al alumno, por lo que el resultado de esa operación se va a encontrar en el papel escrita por él o ella. Sin embargo, será necesario disponer de un botón de manera que al presionarlo se almacene el ejercicio en la aplicación para que el profesor pueda posteriormente evaluar si se ha realizado correctamente o no tanto dicho ejercicio como la autoevaluación del mismo.

DEFINICIÓN DEL PROYECTO

En este capítulo se va a realizar una exposición detallada acerca del proyecto como tal a desarrollar. Para ello, se describirá cómo se encontraba el proyecto en el momento del comienzo del presente trabajo; y en base a este estado se indicarán los objetivos a cumplir y la funcionalidad requerida, que será plasmada en el análisis de los requisitos efectuado previo al comienzo del desarrollo. Finalmente, concretaremos el comportamiento de cada una de las actividades que conformarán la aplicación; con las decisiones tomadas (previo análisis de las posibilidades existentes) en relación con la implementación de todo lo expuesto durante el capítulo.

3.1. Estado Inicial de la Aplicación

Previo al comienzo del proceso tanto de diseño como de desarrollo, es necesario determinar el estado en el que se encuentra la herramienta. El conjunto de las figuras a las que se va a hacer referencia a lo largo de la presente sección se encuentran en el Apéndice C.

Con el fin de poder cubrir toda la aplicación, y poder determinar de manera adecuada, ordenada y completa el estado en el que se encuentra la misma; se ha organizado esta sección en las diferentes vistas o actividades de los que está compuesta esta versión de la aplicación, además del menú principal de esta.

Menú principal

La figura C.1 muestra la vista correspondiente de la actividad del menú principal; tanto con orientación vertical (figura C.1(a)) como horizontal (figura C.1(b)), la cual contiene un total de tres botones. Con estos tres botones se puede navegar a los tres subsistemas de los que se compone:

- Corrige tus ejercicios.
- Ejercicios.
- Ajustes.

Corrige tus ejercicios

En el momento en el que la vista correspondiente a esta actividad es mostrado se inicializa la cámara externa del dispositivo. Se muestran además dos botones, como se aprecia en la figura C.2: el que se encuentra en la esquina superior izquierda permite volver al menú principal, mientras que el situado en la esquina superior derecha muestra una ayuda al usuario en relación con el marcador de Realidad Aumentada y la ficha del ejercicio.

En el momento que la cámara capta el marcador de la Realidad Aumentada y lee los datos contenidos en el código QR que se encuentra en su interior, el botón de ayuda se oculta, y se muestra, de acuerdo con la figura C.3, el resultado de la operación correspondiente adecuadamente situado.

Ejercicios

La vista de los ejercicios creados puede encontrarse en dos estados posibles, que se corresponden con el número de ejercicios que existen en la aplicación: vacía y mayor que cero. En caso de que no haya ejercicios creados, se le indicará tal efecto al usuario como se muestra en la figura C.4(a); mientras que si existen ejercicios se mostrarán estos junto a la fecha de creación y un *checkbox* que permite seleccionarlos y realizar determinadas acciones con ellos, como se explicará a continuación; de acuerdo con la figura C.4(b).

Se dispone además en la parte inferior derecha de un *Floating Action Button*; tal que al presionarlo permite determinar el tipo de la nueva operación aritmética básica (suma, resta, multiplicación y división) a generar (véase la figura C.5). Una vez seleccionada, se genera un diálogo análogo al presente en la figura C.6; de manera que una vez introducidos ambos operandos se genera la operación, actualizando la interfaz.

Cuando se selecciona uno o más ejercicios de entre los existentes, tal como se muestra en la figura C.7 se le indica al usuario el número total de ejercicios seleccionados; y se le ofrecen tres posibilidades:

- 1.– Seleccionar Todo: permite seleccionar todos los ejercicios.
- 2.– Imprimir: imprime los ejercicios seleccionados, generando una ficha como la de la figura C.8 para cada uno de los ejercicios, permitiendo al usuario almacenarla en formato PDF en el almacenamiento del dispositivo.
- 3.– Eliminar: elimina el conjunto de ejercicios seleccionados de la aplicación.

Ajustes

La vista de esta actividad se muestra en la figura C.9. Esta vista tan sólo contiene el ajuste correspondiente con la resolución de la cámara. Seleccionando dicha opción, se despliega un diálogo que permite al usuario determinar la resolución de cámara deseada en base al conjunto de resoluciones disponible, como muestra la figura C.10. Esta resolución será la que se emplee cuando se inicie la cámara en Corrige tus Ejercicios.

3.2. Estructura Lógica

Las diferentes funcionalidades de la aplicación se han estructurado en un total de cinco subsistemas, como se puede ver en la figura 3.1, con la finalidad de mantenerlas lo más organizadas posibles y facilitar su comprensión.

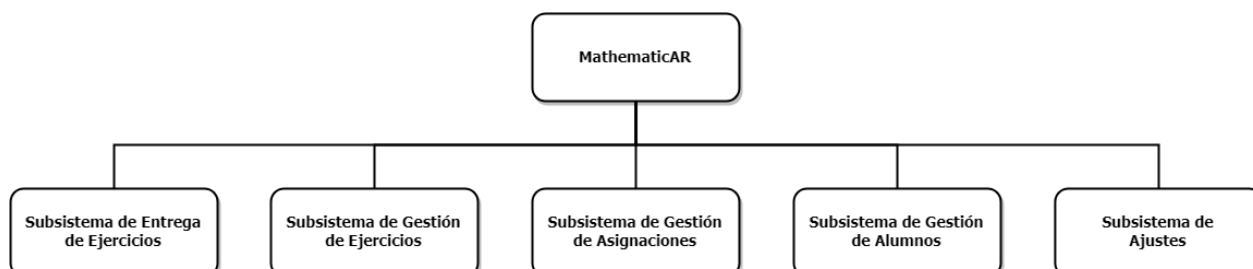


Figura 3.1: Subsistemas de la aplicación MathematicAR

La división del sistema en subsistemas es por tanto la siguiente:

1.– Subsistema de Entrega de Ejercicios.

Se agrupan las funcionalidades relativas a la Realidad Aumentada y la realización de las entregas de las fichas de ejercicios y su correspondiente autoevaluación por parte de los estudiantes.

2.– Subsistema de Gestión de Ejercicios.

Engloba las funcionalidades encargadas de la gestión de los ejercicios de la aplicación; incluyendo la creación, asignación e impresión de los mismos; así como del proceso de eliminación.

3.– Subsistema de Gestión de Asignaciones.

El subsistema de gestión de asignaciones es el encargado de administrar el estado de las entregas de ejercicios de cada uno de los alumnos en sus respectivas sesiones, permitiendo visualizar aquellos que han sido entregados.

4.– Subsistema de Gestión de Alumnos.

Las características relacionadas con la gestión del alumnado; como son la creación, edición y eliminación de estos se agrupan en este subsistema.

5.– Subsistema de Ajustes.

En el subsistema de ajustes se engloban las funcionalidades relativas a la configuración disponible en la aplicación.

3.3. Análisis de Requisitos

En el análisis de los requisitos del proyecto a desarrollar procederemos a desglosar las diferentes funcionalidades que se han de incorporar al mismo. Estos requisitos funcionales se van a organizar en función de los cinco subsistemas que conforman la aplicación. Sin embargo, dado que sobre el Subsistema de Ajustes no se van a definir nuevas funcionalidades, este no ha sido añadido.

Posteriormente, se expondrán los requisitos no funcionales que deberá verificar la aplicación para el cumplimiento satisfactorio de los objetivos planteados.

3.3.1. Requisitos Funcionales

Subsistema de Entrega de Ejercicios

- RF-1.**— Los alumnos podrán realizar entregas de las diferentes fichas de ejercicios que dispongan.
- RF-2.**— Para realizar la entrega, será necesario que el alumno realice una autoevaluación de la entrega realizada; indicando si el resultado de la operación aritmética del ejercicio entregado es correcta o no.
- RF-3.**— Al realizar una entrega, se almacenará en la aplicación la imagen de la ficha tomada por el alumno.
- RF-4.**— No debe ser posible realizar la entrega de una ficha que ya haya sido entregada previamente. En tal caso, se le hará saber este hecho al usuario.
- RF-5.**— La autoevaluación solo puede ser establecida cuando el motor de Realidad Aumentada detecte una nueva ficha.
- RF-6.**— En el proceso de la entrega se debe verificar que el ejercicio de la ficha asignada corresponde con el dispositivo sobre el que se está efectuando la entrega.

Subsistema de Gestión de Ejercicios

- RF-7.**— Permitir la generación de un ejercicio, requiriendo al usuario los siguientes campos:
- Primer operando.
 - Tipo de operación aritmética.
 - Segundo operando.
- RF-7.1.**— Mostrar una vista previa de la solución del ejercicio; previamente a la confirmación de creación del mismo y cuando los operandos de la misma satisfagan las condiciones para poder ser creados.
- RF-8.**— Permitir la generación automática de operaciones aritméticas, requiriendo al usuario los siguientes campos:
- Subtipo de la operación (véanse RF-8.1 y RF-8.2)
 - Número de operaciones que se desean generar.
 - Número de cifras del primer operando.
 - Número de cifras del segundo operando.
- RF-8.1.**— Las sumas y las restas disponen de los siguientes subtipos de operación:
- Sin restricciones.
 - Sin llevada.
 - Con llevada.
- RF-8.2.**— Las multiplicaciones y las divisiones tan solo disponen del subtipo de operación 'Sin restricciones'.
- RF-9.**— Un ejercicio puede ser eliminado de la aplicación.
- RF-9.1.**— La eliminación de un ejercicio de la aplicación implica indirectamente la eliminación de todas las asignaciones que lo hagan referencia.
- RF-10.**— Mostrar los ejercicios organizados en bloques, de acuerdo con los tipos de operaciones aritméticas contemplados en la aplicación: suma, resta, multiplicación y división.
- RF-11.**— Mostrar, en cada uno de los bloques de ejercicios de acuerdo con RF-10, las operaciones ordenadas de manera descendente de acuerdo con la fecha de creación de las mismas.

RF-12.— La interfaz debe ser capaz de mostrar claramente que los tipos de operación aritmética existentes contienen a los ejercicios cuyo tipo de operación sea el que corresponde; y de manera análoga, que los ejercicios contienen a todos los alumnos a los que se les ha asignado dicho ejercicio.

RF-13.— Al presentar al usuario los datos asociados a este subsistema, se mostrará la siguiente información:

RF-13.1.— En caso de ser uno de los tipos de operaciones aritméticas, se mostrará el tipo de operación y el número de ejercicios que contiene.

RF-13.2.— En caso de ser un ejercicio, se mostrará la operación en sí (primer operando, símbolo de la operación y segundo operando); así como la fecha de creación del mismo (incluyendo día y hora) y el número de alumnos a los que dicho ejercicio ha sido asignado.

RF-13.3.— En caso de ser un estudiante, se indicarán el nombre y los apellidos de este, seguido del identificador y el día en el que la asignación del ejercicio fue realizada.

RF-14.— En relación con RF-12 y RF-13, permitir expandir y colapsar los tipos de operación y los ejercicios, facilitando así la interacción del usuario con el sistema.

RF-14.1.— Al expandir o colapsar un elemento que representa un ejercicio, se mostrarán u ocultarán todos los alumnos que tengan ese ejercicio asignado.

RF-14.2.— Al expandir o colapsar un elemento que representa los tipos de operaciones, se mostrarán u ocultarán todos los ejercicios existentes, y por ende, todos los alumnos asociados con dichos ejercicios (en caso de encontrarse alguno expandido).

RF-15.— El usuario podrá realizar búsquedas introduciendo la cadena de texto que desea buscar. La búsqueda se realizará tanto en los ejercicios como en los alumnos, indistintamente.

RF-15.1.— Un resultado de búsqueda se considera coincidente en los ejercicios cuando hay alguna coincidencia con la terna formada por los operandos y la operación.

RF-15.2.— Un resultados de búsqueda se considera coincidente en el caso de los alumnos cuando hay alguna coincidencia con el nombre o los apellidos del mismo.

RF-15.3.— Las subcadenas coincidentes se resaltarán, mientras que los elementos en los que no existan coincidencias no serán mostrados.

RF-16.— Si en los resultados de las búsquedas de ejercicios o alumnos a los que se ha asignado operaciones no se encuentra ningún resultado, se hará saber al usuario de este hecho.

RF-17.— La interfaz debe ser capaz de permitir seleccionar o bien ejercicios o bien alumnos con ejercicios asignados; pero en ningún caso debe permitir seleccionar ambos conjuntamente.

RF-18.— En los ejercicios seleccionados de acuerdo con RF-17, se ofrecerá la posibilidad al usuario de asignar dichos ejercicios a un conjunto determinado de alumnos.

RF-19.— El proceso de impresión consistirá en generar un documento PDF en el que cada una de sus hojas consistirá en una ficha de un ejercicio.

RF-20.— Al proceder con la impresión de un conjunto de fichas de ejercicios, la aplicación debe diferenciar, en relación con RF-17, el conjunto de elementos que se encuentra seleccionado.

RF-20.1.— El proceso de impresión para un ejercicio genera fichas para cada uno de los alumnos que lo tienen asignado.

RF-20.2.— El proceso de impresión para un alumno genera una única ficha, de acuerdo con el ejercicio que tiene asignado.

RF-21.— La ficha de ejercicios debe incorporar, con respecto a la ficha que se disponía al comenzar con el desarrollo del trabajo, los siguientes elementos:

- Nombre y apellidos del alumno.

- Identificador del alumno.
- Fecha de asignación al alumno correspondiente del ejercicio en cuestión.

Subsistema de Gestión de Asignaciones

RF-22.– La organización de los elementos que conforman la interfaz deben seguir la siguiente estructura basada en niveles de profundidad:

- En la profundidad más superficial se encuentra el alumno.
- A profundidad intermedia se encuentran las sesiones en las que han sido asignados los ejercicios al alumno de la profundidad superior.
- A mayor profundidad, se tiene el ejercicio en cuestión.

RF-23.– Al presentar al usuario los datos asociados a este subsistema, se mostrará la siguiente información:

RF-23.1.– En caso de ser un estudiante, se indicarán el nombre y los apellidos de este, seguido del identificador y el número de sesiones en las que se le han asignado ejercicios.

RF-23.2.– En caso de ser una sesión, se mostrará el día correspondiente con la asignación del ejercicio y el número de ejercicios asignados en esa fecha a ese alumno que contiene.

RF-23.3.– En caso de ser un ejercicio, se mostrará la operación en sí (primer operando, símbolo de la operación y segundo operando); así como el estado de entrega de este. En caso de que el estado de entrega sea 'Entregado', se mostrará además la fecha (formada por la fecha y la hora) en la que se efectuó la entrega.

RF-24.– El usuario podrá realizar búsquedas introduciendo la cadena de texto que desea buscar. La búsqueda se realizará en todos los niveles de profundidad, indistintamente.

RF-24.1.– Un resultados de búsqueda se considera coincidente en los alumnos cuando hay alguna coincidencia con el nombre o los apellidos del mismo.

RF-24.2.– Un resultado de búsqueda se considera coincidente para las sesiones cuando hay alguna coincidencia con alguno de los días de esta.

RF-24.3.– Un resultado de búsqueda se considera coincidente en el caso de los ejercicios cuando hay alguna coincidencia con la terna formada por los operandos y la operación.

RF-24.4.– Las subcadenas coincidentes se resaltarán, mientras que los elementos en los que no existan coincidencias no serán mostrados.

RF-25.– Si la búsqueda no arroja ningún resultado en ninguno de los niveles, se hará saber al usuario de este hecho.

RF-26.– La interfaz no debe permitir la selección de los elementos que la conforman, a excepción de los elementos de mayor profundidad (los ejercicios).

RF-26.1.– Al seleccionar un ejercicio, si este se encuentra entregado se mostrará la imagen asociada con dicha entrega.

RF-27.– La única acción posible en relación con los elementos que forman parte de la interfaz respecto a RF-22 es la de expandir o colapsar, de acuerdo con la estructura de elementos especificada.

Subsistema de Gestión de Alumnos

RF-28.– Permitir la creación de un alumno, requiriendo al profesor los siguientes campos:

- Nombre.
- Apellidos.

RF-28.1.— Permitir la creación de un alumno que posea el campo de apellidos vacío.

RF-28.2.— Permitir la creación de un alumno que posea el mismo nombre y los mismos apellidos que uno existente.

RF-29.— Un alumno puede ser eliminado de la aplicación.

RF-29.1.— La eliminación de un alumno de la aplicación implica indirectamente la eliminación de todas las asignaciones que lo hagan referencia.

RF-30.— Los campos del alumno pueden ser editados posteriormente a su creación.

RF-30.1.— No es posible llevar a cabo múltiples ediciones simultáneas.

RF-31.— Al presentar al usuario los datos asociados a este subsistema, se mostrará la siguiente información, relativa al estudiante:

- Nombre.
- Apellidos.
- Identificador.

RF-32.— El usuario podrá realizar búsquedas introduciendo la cadena de texto que desea buscar.

RF-32.1.— Un resultado de búsqueda se considera coincidente cuando existe alguna coincidencia con el nombre o los apellidos del mismo.

RF-32.2.— Las subcadenas coincidentes se resaltarán, mientras que los elementos en los que no existan coincidencias no serán mostrados.

RF-33.— Si en los resultados de las búsquedas no se encuentra ningún resultado, se hará saber al usuario de este hecho.

RF-34.— La interfaz debe ser capaz de permitir seleccionar los alumnos.

RF-35.— En los alumnos seleccionados de acuerdo con RF-34, se ofrecerá la posibilidad al usuario de seleccionar todos los alumnos, editar un alumno y eliminarlos.

RF-35.1.— Al seleccionar más de un alumno, la opción de editar debe ocultarse.

3.3.2. Requisitos No Funcionales

RNF-1.— El dispositivo sobre el que se ejecuta la aplicación debe disponer de al menos una cámara. Si no es así, cuando se inicie la aplicación se impedirá el uso de la misma por este motivo.

RNF-2.— La aplicación debe contar con manuales de usuario, tanto para alumnos como para profesores, estructurados correctamente, de manera que el tiempo de aprendizaje del sistema por parte de un usuario sea inferior a una hora.

RNF-3.— La aplicación debe proporcionar mensajes de error que sean informativos y orientados al usuario final.

RNF-4.— La aplicación tiene que ser estable y manejar que no se den errores en su uso. En caso de producirse algún error, el sistema debe recuperarse frente a ellos sin producirse pérdida de datos.

RNF-5.— Desarrollar un sistema flexible y fácilmente mantenible, de tal forma que sea sencillo realizar futuros cambios (como los indicados en la Sección 6) y corregir posibles problemas. La aplicación debe ser desarrollada de tal forma que cualquier desarrollador pueda añadir nuevas funciones al código fuente de forma sencilla.

RNF-6.— La aplicación será evaluada por el equipo docente de manera regular en el tiempo, de manera que sea posible detectar posibles fallos, así como introducir mejoras sugeridas por los mismos.

RNF-7.— La interfaz gráfica debe ser simple, intuitiva y fácil de utilizar; de forma que cualquier tipo de usuario sepa usarla sin problemas. Además, todos los elementos y pantallas que conforman la interfaz gráfica deberán

mantener un estilo y colores similares.

RNF-8.— La aplicación mostrará mensajes de error en caso de que la aplicación falle durante su ejecución. Estos mensajes deberán ser comprensibles para cualquier tipo de usuario.

RNF-9.— Toda funcionalidad de la aplicación y transacción de negocio debe tener un tiempo de respuesta inferior a los 5 segundos. Si un usuario tiene que esperar un largo tiempo durante alguna de las acciones que está realizando, se deberían mostrar elementos gráficos que adviertan al usuario de la espera.

3.4. Diseño de la Interacción del Sistema

A continuación, en esta sección se detallará el diseño conceptual y visual de las diferentes vistas de la versión final de la aplicación desarrollada; y las posibles interacciones a realizar por parte de un usuario con las mismas. Para ello, se hará uso de capturas de pantalla correspondientes con el estado final de la aplicación, incluidas en el Apéndice D a modo de figuras; a las cuales a lo largo de la presente sección se hará referencia.

La sección es análoga a la sección 3.1, en la cual se especificaba el estado en el que se encontraba previa al comienzo del desarrollo la aplicación; salvo que en este caso se especificará el estado al final del proceso de desarrollo de la misma.

Menú principal

Al acceder a la aplicación el usuario, durante el proceso de carga del menú principal, visualizará una minimalista vista que contiene el icono y el nombre de la aplicación como el de la figura D.1. Una vez la vista del menú principal esté disponible, esta se visualizará.

La figura D.2 muestra la vista correspondiente de la actividad del menú principal; tanto con orientación vertical (figura D.2(a)) como horizontal (figura D.2(b)), la cual contiene un total de cinco botones. Al pulsar sobre alguno de estos botones, se mostrará al usuario la vista correspondiente:

- Corrige tus ejercicios.
- Ejercicios.
- Asignaciones.
- Alumnos.
- Ajustes.

El menú principal muestra además, en su parte superior el nombre de la aplicación; mientras que en la parte inferior se indica la versión instalada en el dispositivo.

Corrige tus ejercicios

En el momento en el que esta vista se inicia, se inicializa la cámara externa del dispositivo. Se muestran además dos botones, como se aprecia en la figura D.3. Presionando sobre el botón que se

encuentra en la esquina superior izquierda se retorna al menú principal, previa consulta al usuario que debe confirmar si realmente desea salir de la cámara (figura D.4); mientras que seleccionando el botón situado en la esquina superior derecha se muestra al usuario un mensaje de ayuda que le indica que debe enfocar a la ficha con la cámara para poder obtener la solución de la misma y autoevaluarse (figura D.5).

En el momento que la cámara capta el marcador de la Realidad Aumentada (véase figura D.6(a)) y lee los datos contenidos en el código QR que se encuentra en su interior, el botón de ayuda se oculta mostrando, de acuerdo con la figura D.6(b), el resultado de la operación correspondiente adecuadamente situado; además de dos nuevos botones en la parte inferior derecha, que permitirán la autoevaluación.

Seleccionando el botón de autoevaluación correcta, este se coloreará en verde, manteniéndose el color original en el botón de autoevaluación incorrecta; y seleccionando el botón de autoevaluación incorrecta, este se coloreará en rojo, manteniéndose el color original en el botón de autoevaluación correcta.

Cuando el usuario presiona alguno de estos dos botones asociados a la autoevaluación, la aplicación procede a almacenar una copia de la ficha haciendo uso de la cámara. Para poder asegurar que la imagen guardada es correcta, el usuario debe mantener la cámara apuntando a la ficha hasta que se le muestre un mensaje de que la entrega se ha efectuado satisfactoriamente (figura D.7(a)).

En caso de producirse algún error, asociado con la ficha; no se mostrará el mensaje indicado anteriormente. En su lugar, se mostrarán mensajes que indiquen tal hecho como se muestra en las figuras D.7(b) (entrega ya realizada) y D.7(c) (ejercicio inexistente).

Ejercicios

Al mostrar esta vista, el usuario puede encontrarse con dos posibles escenarios:

- Si no existen ejercicios creados en el sistema, se muestra la vista de la figura D.8(a).
- Si existen ejercicios, la vista será análoga a la mostrada en la figura D.8(b); donde cada uno de los tipos de operaciones contendrá los ejercicios existentes de su mismo tipo.

En ambos escenarios, la parte superior derecha y superior izquierda de la vista son comunes: la zona superior izquierda muestra el nombre de la actividad y una flecha de dirección que permite al usuario el retorno al menú principal; y la zona superior derecha que contiene las opciones de búsquedas y expansión/colapso de los elementos de la vista (si es que los hubiere).

Seleccionando la opción de búsquedas, la vista se muestra de acuerdo con la figura D.9(a); y el funcionamiento de esta es de acuerdo con lo indicado en la especificación de requisitos de la sección 3.2: si hay coincidencias, ya sea bien en ejercicios (figura D.9(c)) o bien en alumnos (figura D.9(d)) las coincidencias se resaltan; mientras que si no hay coincidencias se le hace saber al usuario (figura

D.9(b)).

También en ambas posibles vistas se dispone de un *Floating Action Button* en la parte inferior derecha; que al presionarlo permite determinar si se genera una nueva operación aritmética básica o un conjunto de operaciones (figuras D.10). En ambos casos, si el usuario selecciona alguna de las dos opciones se mostrarán el conjunto de operaciones que desea realizar: suma, resta, multiplicación o división (figuras D.11(a) y D.11(b), respectivamente).

En el caso de una operación, se mostrará un diálogo como el de la figura D.13(a) según el tipo de operación. Una vez el usuario rellene ambos operandos; seleccionando el botón de 'Ver Vista Previa', se mostrará en la parte derecha del diálogo la solución de dicho ejercicio como en la figura D.13(b). Seleccionando 'Crear' se creará el ejercicio en cuestión; y con 'Cancelar' se mostrará de nuevo la vista de Ejercicios y el ejercicio no se habrá creado.

En el caso de un conjunto de operaciones, el diálogo mostrado es diferente, y se muestra en la figura D.12. El usuario puede determinar el tipo de operación, el número de operaciones a generar y las cifras de los operandos que la componen. Seleccionando 'Crear' se crearán los ejercicios en cuestión; y con 'Cancelar' se mostrará de nuevo la vista de Ejercicios y el conjunto no se habrá generado.

Presionando sobre los tipos de operaciones, estos serán colapsados o expandidos según corresponda como muestran las figuras D.14(a) y D.14(b). En caso de no contener ejercicios, no realizará ninguna acción. Presionando prolongadamente sobre los ejercicios o los alumnos asignados a los ejercicios, estos serán seleccionados; de igual manera sucede pulsando los iconos asociados a los mismos. En el caso de los ejercicios, la presión normal tendrá un comportamiento análogo al de los tipos de operaciones (véase figura D.14(c)). La figura D.15 muestra además como, cuando un alumno o ejercicio son seleccionados, la vista muestra en la zona en la que se mostraba el nombre de la actividad el número de elementos seleccionados.

Cuando se selecciona uno o más ejercicios de entre los existentes (análogamente con los estudiantes), tal como se muestra en la figura D.16(a) (análogamente la figura D.16(b)), se le ofrecen cuatro o tres posibilidades, respectivamente:

- 1.– Seleccionar Todo: permite seleccionar todos los elementos del mismo tipo que los que ya se encuentran seleccionados, ya sean ejercicios o estudiantes.
- 2.– Imprimir: imprime las fichas de ejercicios. Si la selección son ejercicios, se imprimen las fichas de todos los alumnos que tengan asignados esos ejercicios (el proceso lo muestran las figuras D.17(a) y D.17(b)); mientras que si la selección es un alumno tan sólo se imprime las fichas correspondientes a esas asignaciones (el proceso se muestra en las figuras D.18(a) y D.18(b)). Al seleccionar esta opción, se genera una ficha análoga con la de la figura E.1 para cada una de las asignaciones, permitiendo al usuario almacenarla en formato PDF en el almacenamiento del dispositivo.
- 3.– Asignar: muestra la vista correspondiente a los alumnos y permite seleccionar aquellos a los que se desea asignar los ejercicios. Se verá con más detalle en la vista correspondiente a los alumnos.
- 4.– Eliminar: elimina el conjunto de ejercicios o asignaciones a alumnos seleccionados de la aplicación. Una vez

seleccionada esta opción, el usuario visualizará en la parte inferior de la interfaz un mensaje informativo con el número de elementos del tipo correspondiente eliminados, permitiendo durante unos segundos deshacer ese borrado y restaurar el elemento o elementos eliminados.

En relación con las fichas de ejercicios, destacar que dado que ahora estas fichas contienen información no sólo del ejercicio en sí, sino también de la asignación que lo contiene; se necesita saber a quién se ha asignado el ejercicio y en qué fecha. Para ello, las fichas incluyen en su parte superior el alumno al que ha sido asignado dicho ejercicio junto con su identificador, y la fecha de asignación del ejercicio en cuestión al alumno correspondiente.

Si el usuario hace un *swipe* hacia la izquierda (figura D.19(a)) o la derecha (figura D.19(b)) en los ejercicios o alumnos, se efectuará el borrado de dicho elemento, mostrando el mismo mensaje que si eliminaran con la opción de borrado descrita anteriormente, como se indica en la figura D.20.

Asignaciones

En el momento en el que esta vista es creada, el usuario puede encontrarse con dos posibles escenarios:

- Si no existen o bien alumnos o bien asignaciones creadas en el sistema, se muestra la vista de la figura D.21(a).
- Si existen asignaciones, la vista será análoga a la mostrada en la figura D.21(b).

En ambos escenarios, la parte superior derecha y superior izquierda de la vista son comunes: la zona superior izquierda muestra el nombre de la actividad y una flecha de dirección que permite al usuario el retorno al menú principal; y la zona superior derecha que contiene las opciones de búsquedas sobre los elementos de la misma.

Seleccionando la opción de búsquedas, la vista se muestra de acuerdo con la figura D.22(a); y el funcionamiento de esta es de acuerdo con lo indicado en la especificación de requisitos de la sección 3.2: si hay coincidencias, ya sea bien en alumnos (figura D.23(a)), sesiones (figura D.23(b)) o bien en asignaciones (figura D.23(c)) las coincidencias se resaltan; mientras que si no hay coincidencias se le hace saber al usuario (figura D.22(b)). Además, en el caso concreto de esta vista, el usuario no podrá alterar los datos que se muestran en ella.

Presionando sobre los alumnos, estos serán colapsados o expandidos según corresponda. En caso de no contener asignaciones, no realizará ninguna acción. En el caso de los datos de sesión, la presión de este elemento tendrá un comportamiento análogo al caso anterior, salvo que en este caso se mostrarán los datos de los ejercicios asignados al alumno en esa sesión. Al final de este proceso, el estado de la interfaz es similar al mostrado en la figura D.24.

Finalmente, si el usuario presiona sobre un elemento que es un ejercicio; tanto si la entrega del mismo ha sido realizada por el alumno como si no se ha efectuado; se muestra una nueva vista. Esta vista se comportará de manera diferente según el estado en el que se encuentre la entrega:

- Si la entrega se ha realizado, se mostrará la imagen de dicha entrega. Esta imagen contendrá la ficha con su resolución y la autoevaluación efectuada por el alumno (figura D.25(b)).
- Si la entrega no ha sido efectuada, la vista mostrada será análoga a la de la figura D.25(a).

Alumnos

Al mostrar esta vista, el usuario puede encontrarse con dos posibles escenarios:

- Si no existen alumnos creados en el sistema, se muestra la vista de la figura D.26(a).
- Si existen alumnos, la vista será análoga a la mostrada en la figura D.26(b); donde se mostrarán los alumnos con la información de los mismos.

En ambos escenarios, la parte superior derecha y superior izquierda de la vista son comunes: la zona superior izquierda muestra el nombre de la actividad y una flecha de dirección que permite al usuario el retorno al menú principal; y la zona superior derecha que contiene las opciones de búsquedas.

Seleccionando la opción de búsquedas, la vista se muestra de acuerdo con la figura D.27(a); y el funcionamiento de esta es de acuerdo con lo indicado en la especificación de requisitos de la sección 3.2: si hay coincidencias estas son resaltadas como en la figura D.27(c); mientras que si no hay coincidencias se le hace saber al usuario (figura D.27(b)).

También en ambas posibles vistas se dispone de un *Floating Action Button* en la parte inferior derecha; que al presionarlo mostrará un diálogo como el de la figura D.28. Una vez el usuario rellene el nombre y los apellidos (o sólo el nombre); seleccionando el botón 'Crear' se creará el estudiante en cuestión; y con 'Cancelar' se mostrará de nuevo la vista de Alumnos y el alumno no se habrá creado.

Presionando prolongadamente sobre un alumno, este será seleccionado; de igual manera sucede pulsando los iconos asociados a los mismos (figura D.29). Esta figura D.29 muestra además como, cuando un alumno es seleccionado, la vista muestra en la zona en la que se mostraba el nombre de la actividad el número de alumnos seleccionados.

Cuando se seleccionan uno o más alumnos de entre los existentes, se le ofrecen dos o tres posibilidades; mostradas en las figuras D.30(a) y D.30(b):

- 1.– Seleccionar Todo: permite seleccionar todos los estudiantes.
- 2.– Editar: edita los datos (nombre o apellidos) de un alumno. El usuario podrá elegir esta opción cuando el número de estudiantes no sea superior a uno, puesto que las ediciones múltiples no se han contemplado. En este caso, se muestra el diálogo de la creación del estudiante con los datos actuales del mismo, y el comportamiento es análogo a la creación de un alumno nuevo.
- 3.– Eliminar: elimina el conjunto de alumnos seleccionados de la aplicación. Una vez seleccionada esta opción, el usuario visualizará en la parte inferior de la interfaz un mensaje informativo con el número de alumnos eliminados, permitiendo durante unos segundos deshacer ese borrado y restaurar el elemento o elementos eliminados.

Si el usuario hace un *swipe* hacia la derecha o la izquierda en un alumno, se efectuará el borrado de dicho elemento, mostrando el mismo mensaje que si eliminaran con la opción de borrado descrita anteriormente.

Finalmente, si esta vista es mostrada como consecuencia de haber seleccionado la opción de asignación en los Ejercicios, al seleccionar a los alumnos de entre los existente se muestra la opción de la figura D.30(c), de manera que al seleccionarla el ejercicio es asignado a esos alumnos.

Ajustes

La vista de esta actividad se muestra en la figura D.31. Esta vista cuenta tan sólo con el ajuste relativo a la resolución de cámara deseada por el usuario en la vista 'Corrige tus ejercicios'. Seleccionando dicha opción, se despliega un diálogo como el de la figura D.32 que permite al usuario determinar la resolución de cámara deseada en base al conjunto de resoluciones posibles en el dispositivo. Pulsando sobre la opción de 'Cancelar', se mantendrá el valor determinado previamente a la muestra del diálogo.

3.5. Análisis y Propuesta de Solución

A pesar de partir de un proyecto ya existente, será necesario añadir numerosa funcionalidad de acuerdo con lo explicado a lo largo de las secciones previas de este mismo capítulo. En relación con ello, en la presente sección nos enfocaremos en las diferentes vías de desarrollo que se ofrecen para incorporar dicha funcionalidad.

En relación con el Subsistema de Gestión de Ejercicios, una de las funcionalidades a añadir en dicho subsistema es la generación automática de ejercicios. Para ello, la decisión de diseño que se ha tomado en relación con esto es modificar la funcionalidad del Floating Action Button situado en la parte inferior derecha de dicho subsistema, que hasta ahora se encargaba de crear una sola operación.

Así, lo que se requiere ahora es que el Floating Action Button ofrezca la posibilidad al usuario de crear o bien una sola operación o bien esta nueva funcionalidad relativa a un conjunto de operaciones; buscando que, a nivel de interfaz, esta tarea sea lo más sencilla e intuitiva posible. A continuación se muestran las dos alternativas para poder realizar esto.

Las dos alternativas para conseguir esto son *FAB Speed Dial* [6] y *Floating Action Button Speed Dial* [7]. En el caso de la segunda librería, el proyecto está basado en el primero, pero sigue actualizándose y lanzando nuevas versiones (la última *release* es de apenas unos ocho meses); mientras que la primera se ha quedado obsoleta (la última *release* es de casi tres años).

Este el motivo fundamental que ha hecho decantarnos por la segunda, ya que tras efectuar numerosas pruebas con ambas dentro del proyecto, con la primera han surgido numerosos errores de

compatibilidad debido a su falta de actualizaciones; mientras que con la segunda apenas se han sufrido problemas.

Por otro lado, la funcionalidad más crítica de todo el desarrollo realizado en el presente trabajo hace referencia a la interfaz de usuario y a las múltiples interacciones posibles del usuario con la misma. Las alternativas en este caso corresponden con la librería que ya había sido empleada en el desarrollo previo al actual, *Flexible Adapter* [8]; y otras posibles opciones son *Fast Adapter* [9] y *Power Adapters* [10].

Con la última de las tres librerías mencionadas se han obtenido innumerables errores de compatibilidad con las diferentes librerías ya existentes; y con los valores de SDK asignados en el *Gradle*. Por tanto, se ha decidido desechar esta posibilidad.

En cuanto a las dos librerías restantes, tras probar más en profundidad *Flexible Adapter* respecto a la funcionalidad que se emplea en la aplicación; e integrar *Fast Adapter*, se ha comprobado que el funcionamiento de ambas librerías en términos de los elementos de las mismas que nos interesa incorporar al proyecto es muy similar.

Sin embargo, utilizar la librería *Fast Adapter* nos implica tener que rehacer la mayor parte de la funcionalidad que ya se ha cubierto con *Flexible Adapter*; puesto que a pesar de que son similares hay que hacer pequeñas modificaciones en los ficheros implicados en este cambio de librería. Es por ello que finalmente, se ha decidido mantener la librería *Flexible Adapter*, presente en el estado inicial del proyecto software.

En cuanto al resto, se ha tomado la decisión de proseguir en la misma línea con respecto al desarrollo ya realizado previamente en términos de posibles soluciones, puesto que no se ha considerado provechoso el hecho de comenzar con una determinada solución cuando ya se dispone de una que es correcta y funciona adecuadamente. Se ha hecho uso de las soluciones ya existentes, orientándolas a los requerimientos establecidos para la aplicación.

DESARROLLO

La fase del desarrollo de la aplicación se ha llevado a cabo de manera simultánea con la etapa de diseño de la misma efectuado. Dado que, en líneas generales, el desarrollo se ha realizado de manera modular con respecto a los subsistemas que conforman la aplicación, las fases de diseño y desarrollo no se han efectuado de manera consecutiva a nivel global, sino que se han realizado consecutivamente desde el punto de vista de cada uno de los subsistemas.

4.1. Herramientas empleadas

En esta sección se indicarán el conjunto de las herramientas relevantes empleadas durante todo el proceso de desarrollo de la aplicación.

4.1.1. Lenguajes de programación

En relación con los posibles lenguajes sobre los cuales desarrollar una aplicación móvil para dispositivos Android, las principales opciones giran en torno a dos lenguajes de programación: Java y Kotlin; aunque también es posible realizar dicho desarrollo en los lenguajes de programación de C y C++.

Sin embargo, dado que partimos de un desarrollo realizado previamente (véase Sección 3.1); se ha decidido continuar empleando el lenguaje de programación Java, que es el lenguaje en el que se encuentra desarrollada la aplicación inicialmente.

4.1.2. Librerías

Android Architecture Components

Android Architecture Components [11] son un conjunto de librerías que ofrecen ayuda a los desarrolladores en relación con el diseño robusto, comprobable y que permita el mantenimiento de sus aplicaciones. Es usada esencialmente en el manejo del ciclo de vida de los componentes de la interfaz

de usuario y el manejo de datos persistentes. Dentro de dicha librería, se ha hecho uso de las clases:

- **ViewModel**: almacena información relativa a la interfaz de usuario, de manera que mantiene los datos de la misma. Esto es especialmente importante, por ejemplo, para mantener los datos de una actividad o fragmento cuando se produce una rotación de pantalla y manejar las comunicaciones entre actividades y fragmentos.
- **LiveData**: es una clase que se encarga del manejo de datos observables, de manera que cuando se produce una modificación de dichos datos todos los observadores activos son notificados. Esta puntualización es importante, ya que al no notificar a los inactivos, respeta el ciclo de vida de otros componentes de la aplicación.
- **Room**: proporciona una capa de abstracción sobre el sistema de gestión de bases de datos relacional SQLite, permitiendo un acceso más robusto al mismo tiempo que se aprovecha todo el poder del mismo.

EasyAR

EasyAR [12] es un motor de Realidad Aumentada. Es un software gratuito, aunque posee una licencia de pago que permite un mayor número de funcionalidades que en nuestro caso no son relevantes. Sin embargo, es necesario registrarse para poder obtener una *key*; que permita posteriormente inicializar el SDK de EasyAR.

Dicha librería es fundamental para poder mostrar las soluciones de los ejercicios asignados a los alumnos; pues en cada una de las fichas de ejercicios se incorpora el marcador de Realidad Aumentada, en cuyo interior se encuentra un código QR con diferente información relativa a la ficha (generado con la librería que será explicada seguidamente a esta). El marcador es necesario, ya que cuando el motor *EasyAR* lo detecte, se procederá a leer el contenido de dicho código QR. El marcador en cuestión se muestra en la figura 4.1, y como se puede ver posee un espacio en su interior donde se situará el código QR que corresponda.



Figura 4.1: Marcador de Realidad Aumentada de MathematicAR

ZXing

ZXing (Zebra Crossing) [13] es una librería *open-source* que nos permitirá obtener los códigos QR que serán colocados en el interior del marcador de Realidad Aumentada, indicado en la figura 4.1 previa. La librería permite el uso de numerosos formatos de *barcodes* soportados, tanto en dos dimensiones como en una; pero tan sólo se hará uso del formato en dos dimensiones correspondiente con el QR.

OpenGL ES 2.0

La biblioteca *EasyAR* ofrece la opción de utilizar esta librería si desea renderizar entornos tanto en dos dimensiones como en tres dimensiones sobre ella. En nuestro caso, se empleará para renderizar el entorno en 3D correspondiente a la solución de las fichas de ejercicios.

Floating Action Button Speed Dial

Esta librería nos permitirá aumentar la complejidad de determinados Floating Action Button, de tal forma que al presionar sobre ellos se ofrezcan diferentes opciones, de la manera que se muestra en la figura 4.2.

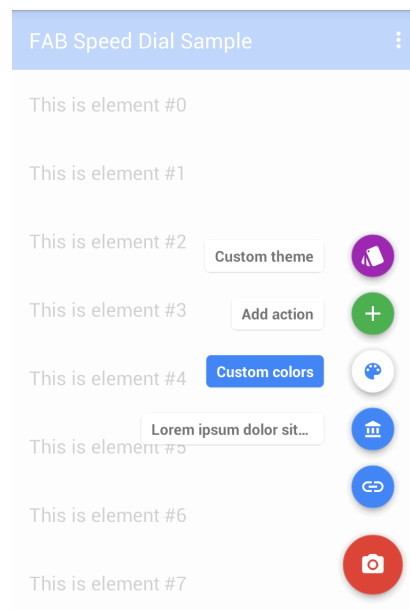


Figura 4.2: Captura de pantalla de un Floating Action Button Speed Dial

Flexible Adapter

Es la librería sobre la cual se sustenta la funcionalidad asociada con la interfaz de usuario y las acciones que este puede realizar dentro de la aplicación. Incluye una cantidad muy elevada de funcionalidades, de las cuales tan sólo se ha hecho uso de una pequeña parte. La finalidad de esta librería es la de intentar aunar diferentes características relativas con la interfaz del usuario, de manera que no sea necesario importar numerosas librerías de terceros; con los problemas de compatibilidad que estas suelen generar entre sí.

Además, intenta simplificar las tareas a realizar por parte del desarrollador facilitando al máximo posible su uso; siendo también extremadamente flexible en cuanto al nivel de configuración de las funcionalidades (es muy sencillo poder añadir o eliminar una de ellas).

4.1.3. Dispositivos

Al comienzo del proceso de desarrollo, este se realizó con una *tablet* ASUS ZenPad C 7.0, con un nivel de API 21.

Tras la primeras reuniones mantenidas con el equipo docente del Colegio de Educación Especial Alenta, dado que las *tablets* de las que disponían poseían niveles de API (además de especificaciones) superiores, se prosiguió con el desarrollo con una *tablet* Samsung Galaxy Tab S4, con un nivel de API 27 en este caso.

4.2. Arquitectura Software

Esta sección contiene la información relativa con la aplicación desarrollada. Se empezará con una perspectiva general de esta, profundizando a medida que se avanza en la misma.

De acuerdo con la división en subsistemas de la aplicación llevada a cabo en la sección 3.2, a partir de la actividad principal de la aplicación (*MainActivity*) se podrá acceder a los cinco subsistemas que la conforman; y cuyas funcionalidades se encuentran claramente diferenciadas. La figura 4.3 representa un diagrama que indica las diferentes posibilidades dentro de la aplicación para transitar de una actividad a otra. Para mayor detalle acerca de estas transiciones, véase la sección 3.4.

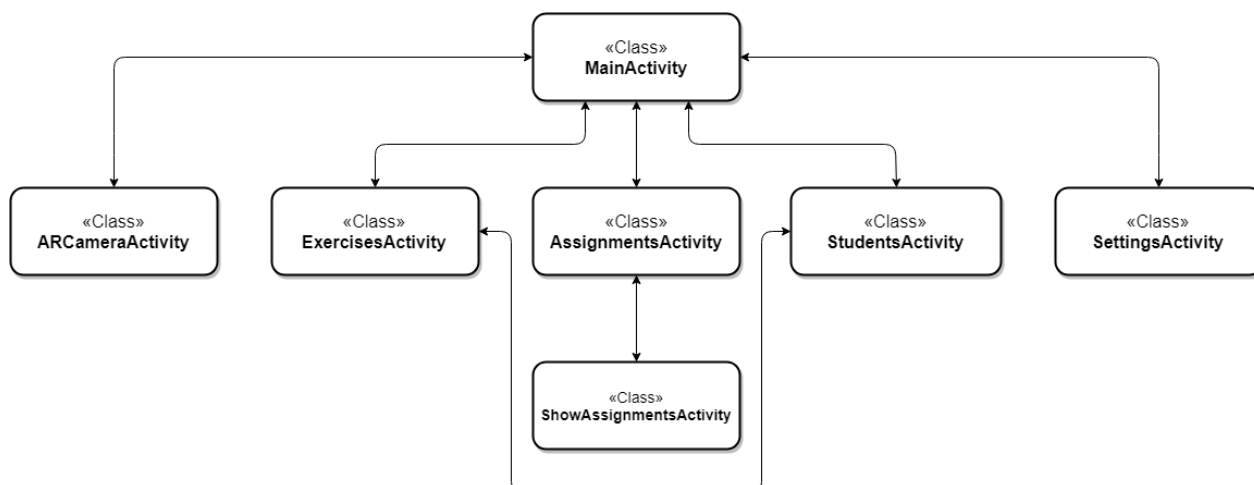


Figura 4.3: Posibles interacciones entre las diferentes vistas de la aplicación

Las flechas del diagrama indican que se puede transitar de una actividad a otra, y viceversa; aunque sólo las que contienen a *MainActivity* lo hacen de manera normal; ya que en el resto de casos se tienen que cumplir ciertos requisitos para que se puedan producir.

En conjunto, estas siete clases Java indicadas constituyen el conjunto de las actividades de la aplicación, cada una de las cuales se corresponde con una Vista de la misma.

A continuación, en las posteriores subsecciones se explicará detalladamente la arquitectura correspondiente a la aplicación MathematicAR, indicando las diferentes interacciones de las actividades con las clases existentes en la aplicación; y que definirán el comportamiento de las mismas en base a lo indicado.

4.2.1. Arquitectura de realidad aumentada y captura de autoevaluación

La Vista de *ARCameraActivity* se encuentra asociada con la Realidad Aumentada y el proceso de la autoevaluación de fichas de ejercicios por parte de los alumnos. La arquitectura de los diferentes componentes que son partícipes en la funcionalidad desarrollada en esta actividad se muestra en la figura 4.4.

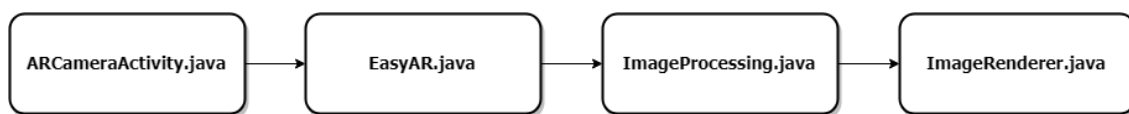


Figura 4.4: Arquitectura de los componentes asociados con la Realidad Aumentada

En el momento en el que se inicia esta actividad, se genera un elemento de la clase *GLSurfaceView*. Esta clase pertenece a la librería *OpenGL ES 2.0*; y será la encargada de mostrar las soluciones a la ficha de un ejercicio que se enfoque con la cámara.

Dicho elemento inicializará a EasyAR; que a partir de ese momento, y hasta que la actividad finalice realizará continuamente la comprobación de qué es lo que se está enfocando con la cámara. Para ello, se distinguen esencialmente tres posibles casos: marcador detectado, código QR detectado e imagen desconocida. Para detectar el código QR será necesario haber detectado previamente el marcador. El contenido del código QR es esencialmente la cadena obtenida con la llamada al método *toQRText()* por parte de un ejercicio (que se explicará más adelante); concatenado con el identificador, que es único, de la asignación correspondiente.

Cada vez que un código QR es reconocido, y este se ajusta a lo esperado, se genera el *bitmap* correspondiente a la solución del ejercicio que está siendo enfocado en la clase *ImageProcessing*, siendo finalmente la clase *ImageRenderer* la encargada de mostrar por pantalla, haciendo uso de la librería *OpenGL ES 2.0* y en la posición que corresponda; dicho *bitmap*.

Cabe destacar el funcionamiento del proceso de guardado de la imagen generada en el proceso de entrega. Dicho proceso se realiza cuando se presiona o bien el botón de autoevaluación correcta, o bien el botón de autoevaluación incorrecta. En ese momento, se toma una imagen del estado actual de la aplicación; incluyendo además estos botones de autoevaluación. Dado que el proceso de captura y procesado de la imagen en cuestión requiere de un par de segundos; una vez el proceso se completa se informa al usuario que ya se ha efectuado la entrega. El fichero de imagen es guardado en el almacenamiento interno de la aplicación, de manera que sólo la aplicación puede acceder a él.

Además, para poder garantizar que no se producen colisiones con los nombres de dichos ficheros; las imágenes de las entregas se almacenan con nombre el identificador de la asignación (que se obtiene al procesar el código QR leído).

4.2.2. Arquitectura de ejercicios

La figura 4.5 muestra la arquitectura asociada con las clases que se ven involucradas en los procesos de gestión de los ejercicios. Dicha arquitectura está basada en el *Model-View-ViewModel Pattern*; que es un patrón basado en intentar alcanzar la mayor separación posible entre la lógica de la aplicación y la interfaz del usuario. Además, dicho patrón representa la arquitectura recomendada por parte de Google para el desarrollo de aplicaciones en Android [14]; y se basa en que cada componente sólo depende de aquel componente que se encuentra un nivel por debajo de él. En el caso del *Exercises-Repository*, se basa en el *Repository Pattern*; patrón que proporciona una abstracción de los datos correspondientes a sus niveles inferiores, de manera que no es necesario lidiar con la configuración de las bases de datos.

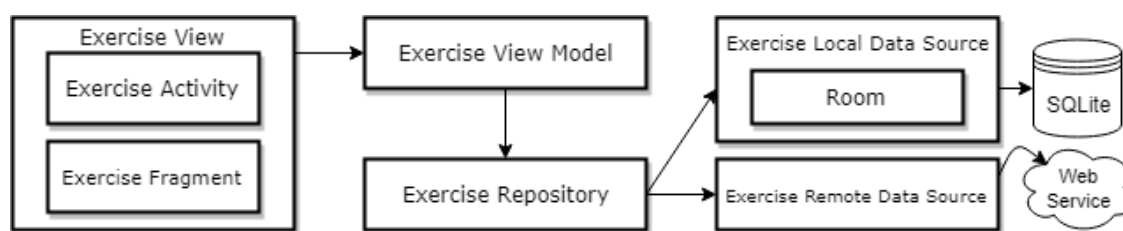


Figura 4.5: Arquitectura de la gestión de los ejercicios

Es por ello que la clase *ExercisesViewModel* tan sólo necesita conocer el conjunto de los métodos que ofrece el repositorio para poder acceder a la base de datos de los ejercicios. En general, dicho repositorio depende tanto de un modelo de datos persistente como de una fuente de datos *backend* remota. Sin embargo, en este caso, no se dispone de la fuente de datos *backend* remota; aunque se ha añadido la funcionalidad necesaria para facilitar esta implementación si fuera necesaria para futuros desarrollos.

4.2.3. Arquitectura de alumnos

La arquitectura relativa a las clases involucradas en la gestión de los alumnos se representa en la figura 4.6; y como se puede ver es completamente análoga a la arquitectura de los ejercicios

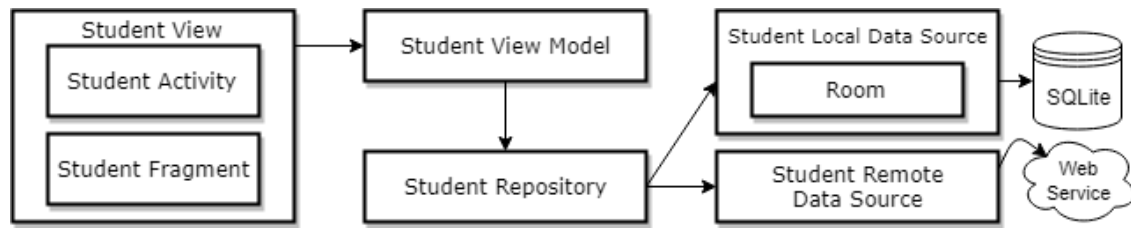


Figura 4.6: Arquitectura de la gestión de los alumnos

4.2.4. Arquitectura de asignaciones

En la figura 4.7 se representa la arquitectura correspondiente con las clases involucradas con la gestión de las asignaciones de ejercicios a alumnos; y al igual que en la arquitectura anterior, es completamente análoga a la arquitectura de los ejercicios.

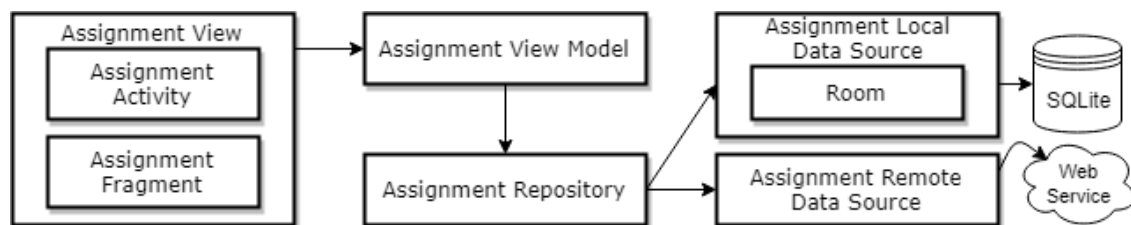


Figura 4.7: Arquitectura de la gestión de las asignaciones

A partir de una instancia de una asignación obtenemos el identificador del ejercicio y del alumno asociado con dicha asignación; por lo que no se dispone de la instancia del ejercicio y del alumno correspondientes. Por ello, con el fin de evitar tener que comparar sobre toda una lista tanto de ejercicios como de alumnos (que además habría que mantener actualizada) elemento a elemento hasta obtenerlo, o de realizar consultas a la base de datos para poder obtenerlo; se va a hacer uso de la anotación *@Relation* que nos ofrece la librería *Room*.

Para ello, se tienen dos clases: *ExercisesWithAssignments* y *StudentsWithAssignments*. Mediante esta anotación se nos va a permitir obtener automáticamente el ejercicio y una lista con los estudiantes a los que se les ha asignado dicho ejercicio; y el estudiante y una lista con los ejercicios que tiene asignados, respectivamente.

4.2.5. Arquitectura de impresión de fichas de ejercicios

En el proceso de impresión de una ficha o conjunto de fichas se encuentran involucrados tanto los *Exercises* como los *Students* y las *Assignments*. La figura 4.8 muestra la arquitectura asociada con este proceso.

La clase *ExercisesPrintAdapter* es la encargada de generar la ficha de la asignación correspondien-

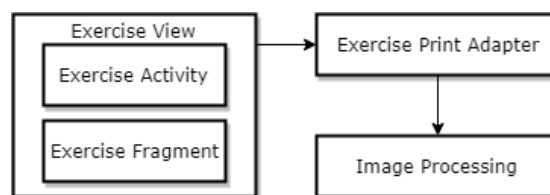


Figura 4.8: Arquitectura para la impresión de fichas

te, con los datos requeridos para la misma; extendiendo de la clase de Android *PrintDocumentAdapter* que se encarga precisamente de tomar un contenido para proceder con la impresión del mismo. La clase *ImageProcessing* se encarga de generar el *bitmap* correspondiente al ejercicio sin su solución.

En términos generales, la impresión consistirá en almacenar el documento en el dispositivo en formato PDF, pero se puede imprimir directamente si está configurada una impresora en el dispositivo. La figura E.1 del Apéndice E muestra una ficha generada por la aplicación.

4.2.6. Arquitectura de la generación automática de ejercicios

El proceso de automatización de los ejercicios es uno de los requisitos en los que el equipo docente ha realizado mayor énfasis; pues les supone una reducción de tiempos importante a la hora de realizar la creación de los ejercicios que posteriormente asignarán. También les permite individualizar el proceso de aprendizaje, de forma que puedan generar numerosos ejercicios conforme a dicho proceso.

Por tanto, en función de la operación aritmética de la que se desee generar un conjunto de las mismas; las acciones a realizar en los dos operandos que conforman la operación diferirán entre unas y otras.

Dado que las operaciones aritméticas de multiplicación y división no tienen ninguna restricción en lo que al proceso de aprendizaje se refiere, la generación de estas se basa en generar números que tengan el número de dígitos especificado por el usuario; controlando además en el caso de la división que el cero nunca es el divisor (segundo operando) de la misma.

En el caso de la suma y la resta, además de la generación de estas operaciones sin ninguna restricción, también existe la posibilidad de generar operaciones con llevada y sin llevada; y en estos casos los procedimientos difieren ligeramente entre una y otra operación:

- Suma con llevada: se generan los dos operandos con el número de dígitos indicado para cada uno de ellos; y se comprueba que la suma contiene al menos una operación que implica una llevada.
- Resta con llevada: se generan los dos operandos con el número de dígitos indicado para cada uno de ellos; y se comprueba que uno de los dígitos del primer operando es menor que el dígito del segundo operando en la misma posición.
- Suma sin llevada: se generan, dígito a dígito, el primer operando; obteniendo en cada una de las iteraciones el

valor del segundo dígito de tal manera que satisfaga que la suma de ambos es inferior a diez.

- Resta sin llevada: el procedimiento es análogo a la suma sin llevada, sólo que el valor que se obtiene para el segundo dígito debe satisfacer que sea menor o igual que el dígito generado para el primer operando.

Además, las comprobaciones tienen en cuenta el hecho muy probable de los dígitos de los operandos no coincidan y, por tanto, uno de los operandos posea más dígitos que otro.

4.2.7. Arquitectura de la interfaz de usuario

Para que la interfaz de usuario se ajuste a los requisitos especificados en la sección 3.3, es necesario que tanto la clase *Exercise* como la clase *Student* extiendan su funcionalidad original e implementen las funcionalidades asociadas con la librería de *Flexible Adapter*, vista en la sección 4.1. Para ello, puesto que además se necesitan ciertos tipos de entidad relativos a esta librería, la figura 4.9 muestra las relaciones de herencia entre las diferentes clases empleadas en relación con la interfaz de usuario.

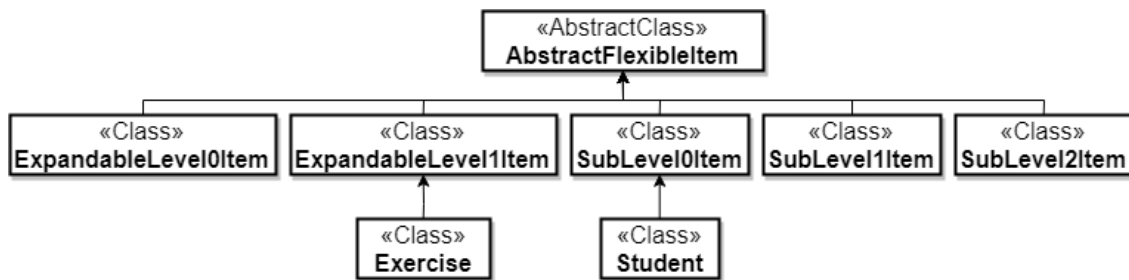


Figura 4.9: Diagrama de las clases implicadas en la interfaz

A continuación, se describe la funcionalidad asociada con cada una de estas clases referente a la interfaz:

- *AbstractFlexibleItem*

Es una clase abstracta de la que heredan todos los ítem relativos a la interfaz. La funcionalidad que contiene está centrada esencialmente en el título y subtítulo de los ítem, además del identificador que debe representar de manera única a un ítem.

- *ExpandableLevel0Item*

Esta clase representa un ítem de la interfaz con profundidad más superficial. Además, contiene una lista de ítem de tipo *ExpandableLevel1Item*. En la aplicación, este ítem representa aquel que contiene los tipos de operación en el Subsistema de Gestión de Ejercicios.

- *ExpandableLevel1Item*

Contiene una lista de ítem de tipo *SubLevel0Item*. La clase *Exercise* hereda de este ítem; y en la aplicación este ítem corresponde con los ejercicios mostrados en el Subsistema de Gestión de Ejercicios.

- *SubLevel0Item*

Contiene una lista de ítem de tipo *SubLevel1Item*. La clase *Student* hereda de este ítem; y en la aplicación este ítem corresponde con los estudiantes asignados a un ejercicio en el Subsistema de Gestión de Ejercicios y en el Subsistema de Gestión de Asignaciones. Mientras que en el primer subsistema es el ítem de mayor profundidad, en el segundo es el más superficial.

- *SubLevel1Item*

En la aplicación, este ítem representa la sesión correspondiente de un estudiante en un ejercicio asignado al mismo. Contiene una lista de ítem de tipo *SubLevel2Item*.

- *SubLevel2Item*

Esta clase representa un ítem de la interfaz de mayor profundidad posible. El ítem dentro de la aplicación se corresponde con la información de la entrega de un determinado ejercicio.

4.3. Modelo de Datos

El conjunto de los datos que se encarga de manejar la aplicación ha cambiado significativamente con respecto a los datos de la etapa inicial del desarrollo.

Ahora, se dispone de ejercicios, alumnos y asignaciones; y los tres se encuentran fuertemente relacionados entre sí. El modelo entidad relación correspondiente se representa en la figura 4.10.

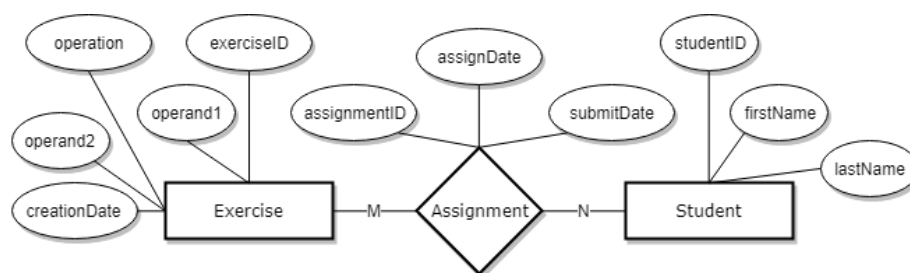


Figura 4.10: Modelo Entidad-Relación de MathematicAR

Por ello, en la base de datos dispondremos de dos entidades, *Exercise* y *Student*. La relación que se establece entre dichas entidades es una relación *many to many* (muchos a muchos). Esto necesariamente implica la existencia de una tercera entidad (una tabla de unión) que una sendas relaciones. Esta entidad se denominará *Assignments*; pues efectivamente las asignaciones de ejercicios requerirán conocer tanto al estudiante al que se le asigna el ejercicio como el ejercicio en sí.

Una instancia de *Exercise* se identifica de manera única mediante un *exerciseID*, y representa una operación aritmética, formada por dos operandos (*operand1* y *operand2*), el tipo de la operación (*operation*) y la fecha de creación de la misma. El tipo de la operación es simplemente una enumeración del conjunto de operaciones permitidas (suma, resta, multiplicación y división). La aplicación no permite generar operaciones tales que sus operandos y operación ya existen; incluso aunque la fecha de creación difiera entre ellas.

Además, el ejercicio dispone del método *toQRText()* correspondiente a la interfaz *QRCodable*. Este método permitirá obtener el string que se almacenará en el código QR de la operación cuando se proceda a su impresión. Esta cadena de caracteres estará formada simplemente por el primer operando, el símbolo de la operación, el segundo operando y el resultado de la misma precedido por un igual.

Por otro lado, una instancia de *Student* es identificada de forma única a través de un identificador, y representa a un alumno que va a utilizar la aplicación. Un estudiante dispone tan sólo de un campo de nombre (*firstName*) y otro de apellidos (*lastName*). En este caso, a diferencia de los ejercicios, se permite la creación de dos alumnos con el mismo nombre y apellidos; y el apellido no es obligatorio, por lo que es posible crear estudiantes tan sólo asignándoles un nombre.

En cuanto a la entidad (tabla) de unión, cada instancia posee un identificador que lo representa de manera única y que además nos será útil en el momento de almacenar la entregas realizadas, para garantizar que el nombre del fichero de imagen no coincide con alguno ya existente. Posee además una fecha de asignación del ejercicio al alumno (*assignDate*), así como la fecha en la que el alumno efectuó la entrega del mismo (*submitDate*). La aplicación no permite realizar la asignación de un ejercicio a un alumno más de una vez en el mismo día, pero sí en días diferentes. De esta forma, un profesor puede reasignar una tarea que se realizó incorrectamente a un alumno múltiples veces hasta realizarla correctamente, con el fin de verificar que efectivamente ha sabido realizarla.

El uso de la anotación *@Relation* de la librería *Room* mencionada en la subsección 4.2.4 es empleada en relaciones de tipo *one to many* (uno a muchos). Por tanto, en este caso, lo que se está haciendo es tratar una relación *many to many* como dos relaciones de tipo *one to many*.

Tanto en el caso de la entidad *Exercise* como *Assignments*, cuando se dice que la aplicación no permite realizar una cierta acción, esto significa que la aplicación no inserta un nuevo registro en la base de datos de la misma. Además, no se notifica al usuario de este hecho por pantalla, puesto que se ha indicado en el manual que estas acciones no tienen ningún efecto.

INTEGRACIÓN, PRUEBAS Y RESULTADOS

En este capítulo se describirá el trabajo realizado durante las diferentes fases de pruebas del proyecto, indicando los diversos tipos de pruebas realizadas, y los resultados y problemas surgidos durante las mismas. En esta fase de pruebas; y en general todas las que componen el proyecto, han sido empleados los dispositivos especificados en la sección 4.1.

Android Studio (el IDE oficial para el desarrollo de aplicaciones Android) proporciona una potente herramienta de emulación de dispositivos a diferentes niveles de API y con numerosas y diversas características; sin embargo, no ha sido posible hacer uso de ella durante el desarrollo de la aplicación pues los diferentes elementos requeridos al terminal lo dificultaban enormemente.

En primer lugar, se realizaron numerosas pruebas unitarias sobre los diferentes módulos que componían la aplicación ya desarrollada; con el fin de poder comprender adecuadamente el funcionamiento de la misma, y poder encontrar algún comportamiento anómalo en esta.

Posteriormente, a medida que se realizaba el desarrollo del software, se han realizado diferentes pruebas asociadas con las funcionalidades especificadas para la aplicación indicadas en la subsección 3.3.1. En dichas pruebas, inicialmente y siempre que fuera posible se ha realizado el desarrollo de la funcionalidad del módulo correspondiente en un proyecto auxiliar; y en caso de no ser posible se ha desarrollado directamente en el proyecto principal.

Una vez realizado el desarrollo correspondiente, ya sea bien en el proyecto auxiliar o bien en el principal, se procedía a realizar las pruebas unitarias del mismo; intentando buscar errores en cada uno de los módulos, no siendo suficiente el funcionamiento correcto del flujo normal de ejecución. En caso de que ocurriera algún error, estos eran corregidos y las pruebas eran ejecutadas de nuevo. El objetivo era poder partir de una base que funcionara según lo esperado, garantizando así el correcto funcionamiento de los módulos previo al proceso de integración de dichos módulos en la aplicación.

Una vez comprobado el correcto funcionamiento del módulo o los módulos en cuestión independientemente, estos eran añadidos al proyecto principal; procediendo posteriormente con la realización de las pruebas de integración. La finalidad de la realización de este tipo de pruebas es garantizar que el conjunto de módulos sigue funcionando correctamente tras dicha integración, y que ninguna de las

funcionalidades asociadas se ve afectada en relación con posibles errores aparecidos.

Sobre estas pruebas destacar, además, como ya se ha mencionado a lo largo del trabajo, que se han realizado reuniones periódicas con el personal docente del Colegio de Educación Especial Alenta; en las cuales a medida que se han realizado avances en el desarrollo se ha requerido validación por su parte para garantizar que esa funcionalidad desarrollada se ajustaba a lo requerido inicialmente.

Por último, se han realizado pruebas en relación con los requisitos no funcionales de la aplicación. Durante todo el proceso de desarrollo del proyecto se ha buscado garantizar el cumplimiento de dichos requisitos, pero es una vez el desarrollo concluye cuando se necesita comprobar que efectivamente ha sido así. Para ello, en aquellos requisitos que requieren de los usuarios de la aplicación, se han aprovechado las últimas entrevistas con los docentes para tal efecto; obteniendo la aprobación por su parte.

CONCLUSIONES Y TRABAJO FUTURO

6.1. Conclusiones

Este trabajo surge debido a la necesidad incorporar nuevas funcionalidades a una aplicación para dispositivos móviles basada en Android ya existente; para que esta se ajuste más fielmente a los procesos de aprendizaje llevados a cabo en las aulas en términos de la educación especial.

Por tanto, se parte de un proyecto ya existente elaborado por otro desarrollador; por lo que previamente al comienzo del diseño y desarrollo en base a dichas nuevas funcionalidades se ha requerido comprender totalmente el funcionamiento de dicha aplicación.

Con estas nuevas funcionalidades, la aplicación ya puede ser empleada en el entorno educativo de manera completa. Los profesores disponen de una herramienta con la podrán administrar de manera sencilla a sus alumnos y asignar diferentes ejercicios basados en operaciones aritméticas a estos (generados manual o automáticamente); mientras que los alumnos, haciendo uso de la Realidad Aumentada, podrán realizar las entregas de dichos ejercicios con su correspondiente autoevaluación a través de la aplicación de manera que su profesor pueda visualizar dicha entrega.

Para poder garantizar el cumplimiento de los objetivos que se persiguen al comienzo del desarrollo del proyecto software, como se ha expuesto a lo largo del presente informe, se han propuesto diferentes soluciones.

El contacto periódico que se ha mantenido con los profesores del centro ha permitido además recoger las peticiones elaboradas por los mismos en relación tanto con funcionalidades ya desarrolladas y disponibles como en aquellas aún por elaborar, incorporándolas al desarrollo. Ello nos ha permitido obtener un producto final que verdaderamente se ajusta a lo solicitado por estos, incorporando además algunas características que en un primer momento no se habían tenido en cuenta.

Con el desarrollo de este trabajo se han aplicado muchos de los conocimientos obtenidos durante los estudios del grado. Sin embargo, también se ha necesitado adquirir otros muchos; especialmente aquellos basados principalmente en la tecnología de la Realidad Aumentada, la cual era prácticamente desconocida a tanto nivel de profundidad previamente al comienzo de este trabajo.

6.2. Trabajo Futuro

En base al estado en el que se encuentra la aplicación una vez concluido su desarrollo en lo que que a este trabajo respecta, aparecen nuevas líneas de futuro desarrollo que permitirán afianzar las funcionalidades existentes (confirmando que se ajustan a lo deseado); así como ampliar dicho conjunto de funcionalidades. Los trabajos futuros considerados son los siguientes:

- Evaluar la aplicación con los usuarios finales (alumnos y profesores del Colegio de Educación Especial Alenta) para obtener retroalimentación de la misma y acometer las pertinentes modificaciones si procede; y poder determinar si efectivamente el uso de estas tecnologías beneficia en el proceso de aprendizaje. Mi tutor se encuentra en conversaciones con este centro para, el curso que viene, realizar un piloto con los alumnos de dicho centro.
- El juego se ha utilizado desde siempre para fomentar el aprendizaje. Por tanto, un trabajo futuro sería emplear la técnica de gamificación [15] junto con la metodología ABA [16]; asociando a cada alumno una moneda virtual con la que comprar personajes que se mostrarán tras realizar un número consecutivo determinado de operaciones aritméticas correctas.

BIBLIOGRAFÍA

- [1] Samsung, "Ocho de cada diez profesores españoles piensa que la tecnología se usa en el aula todas las semanas," tech. rep., 2016.
- [2] A. G. F. de Mendiola, "Aplicación de Realidad Aumentada para favorecer el aprendizaje autónomo de personas con diversidad funcional intelectual," 2018.
- [3] "FETCH! Lunch Rush." <https://pbskids.org/apps/fetch-lunch-rush.html>.
- [4] "AR Flashcards Addition." <http://arflashcards.com/ar-flashcards-addition/>.
- [5] "Math Alive." <https://alivestudiosco.com/math-alive-kit/>.
- [6] Y. Ivanov, "FAB Speed Dial." <https://github.com/yavski/fab-speed-dial>, 2016.
- [7] R. Leinardi, "Floating Action Button Speed Dial." <https://github.com/leinardi/FloatingActionButtonSpeedDial>, 2019.
- [8] D. Steduto, "Flexible Adapter." <https://github.com/davideas/FlexibleAdapter>, 2019.
- [9] M. Penz, "Fast Adapter." <https://github.com/mikepenz/FastAdapter>, 2019.
- [10] NextFaze, "Power Adapters." <https://github.com/NextFaze/power-adapters>, 2019.
- [11] "Android Architecture Components." <https://developer.android.com/topic/libraries/architecture/>.
- [12] "EasyAR." <https://www.easyar.com/>.
- [13] ZXing, "Zebra crossing." <https://github.com/zxing/zxing>, 2019.
- [14] "Arquitectura recomendada para el desarrollo de aplicaciones Android." <https://developer.android.com/jetpack/docs/guide>.
- [15] R. Colpani and M. R. P. Homem, "An innovative augmented reality educational framework with gamification to assist the learning process of children with intellectual disabilities," in *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pp. 1–6, July 2015.
- [16] N. Peters-Scheffer, R. Didden, H. Korzilius, and P. Sturmey, "A meta-analytic study on the effectiveness of comprehensive aba-based early intervention programs for children with autism spectrum disorders," *Research in Autism Spectrum Disorders*, vol. 5, no. 1, pp. 60 – 69, 2011.

APÉNDICES

MANUAL DE USUARIO PARA ALUMNOS

Cuando la aplicación sea iniciada, se mostrará un menú como el indicado en la figura A.1. Si tienes ejercicios realizados y quieres hacer la entrega, selecciona el botón de 'Corrige tus ejercicios'.

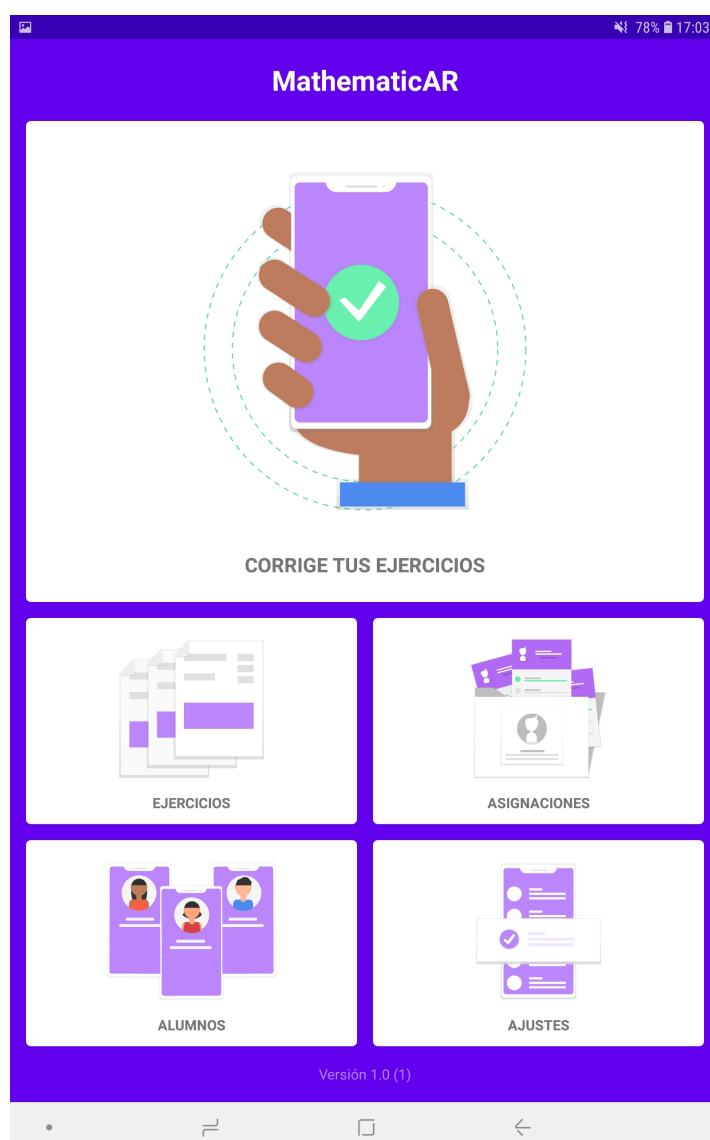
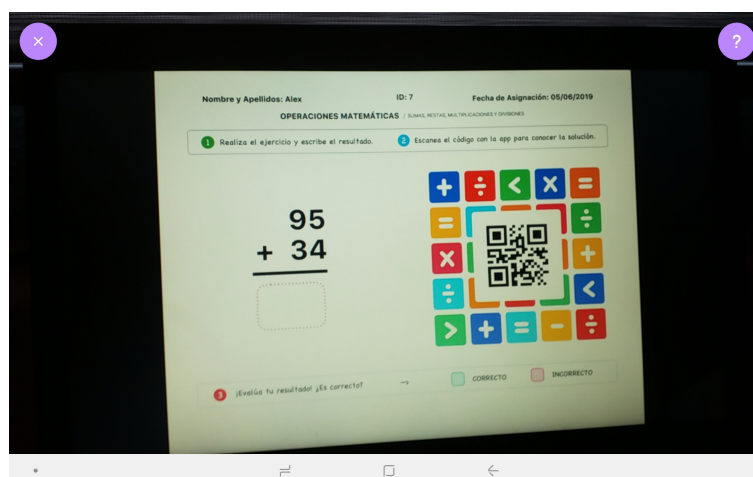
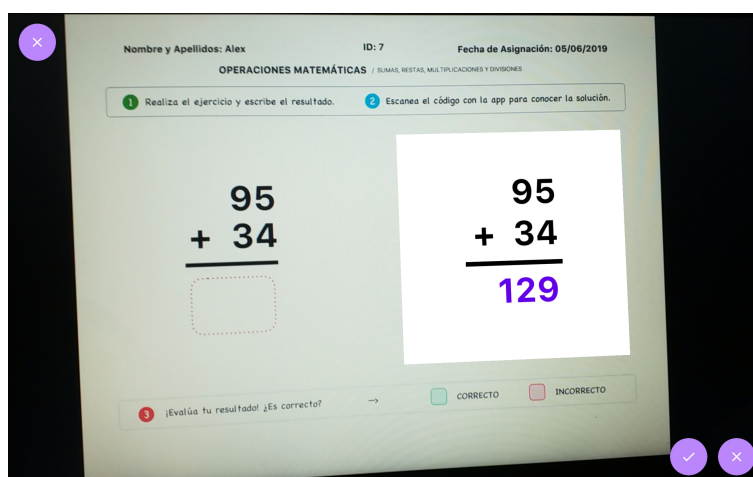


Figura A.1: Menú principal de MathematiCAR

La cámara del dispositivo se abrirá, y ya puedes enfocar los ejercicios a corregir. Si quieres regresar al menú principal o necesitas ayuda, selecciónalo en la pantalla. Una vez se enfoque al ejercicio, se mostrará la solución en el código QR, de acuerdo con el proceso mostrado en la figura A.2.



(a) Ficha sin reconocer



(b) Ficha reconocida

Figura A.2: Ficha enfocada con la cámara

Comprueba el resultado y autoevalúate. Una vez presiones alguna de las dos opciones de autoevaluación, mantén enfocada la ficha del ejercicio hasta que se muestre un mensaje como el de la figura A.3.

Entrega realizada

El ejercicio se ha entregado correctamente.

ACEPTAR

Figura A.3: Mensaje de entrega satisfactoria

¡Ya has realizado una entrega! Si tu profesor te ha asignado más ejercicios, puedes hacerlos y entregarlos si así lo deseas.

La resolución de la cámara se puede modificar en el menú de los ajustes, al que se puede acceder desde el menú principal.

MANUAL DE USUARIO PARA DOCENTES

Cuando la aplicación sea iniciada, se mostrará un menú como el indicado en la figura B.1. Como profesor, tus tareas se enfocan en los botones de 'Ejercicios', 'Asignaciones' y 'Alumnos'.

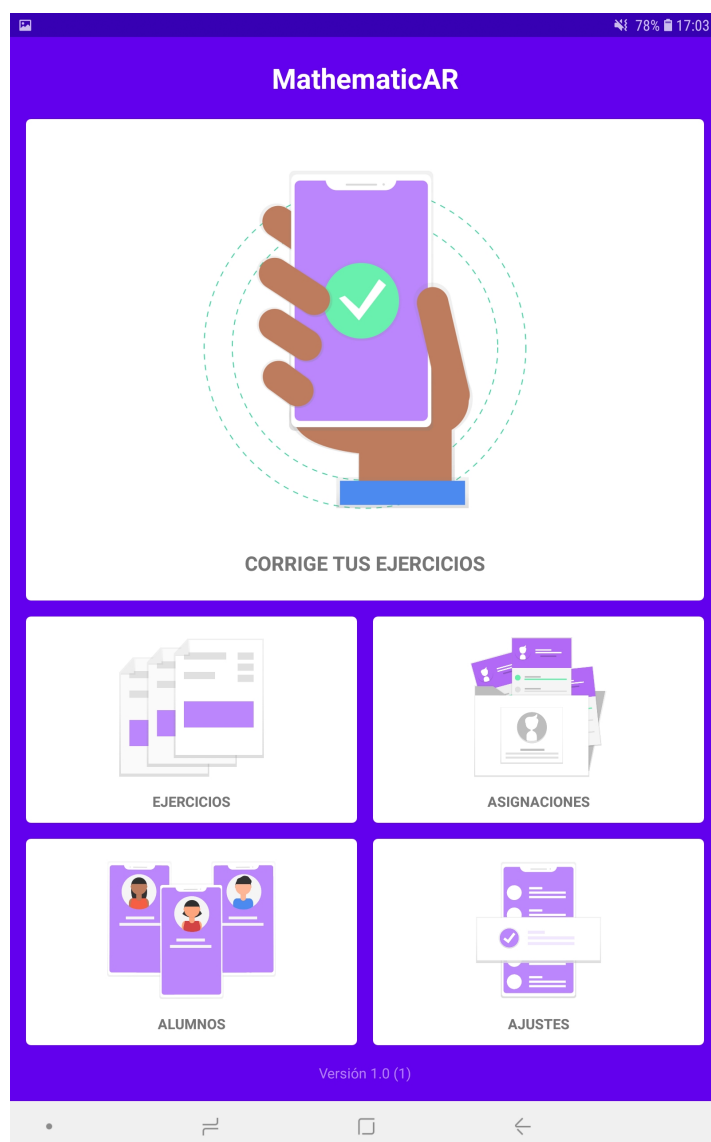


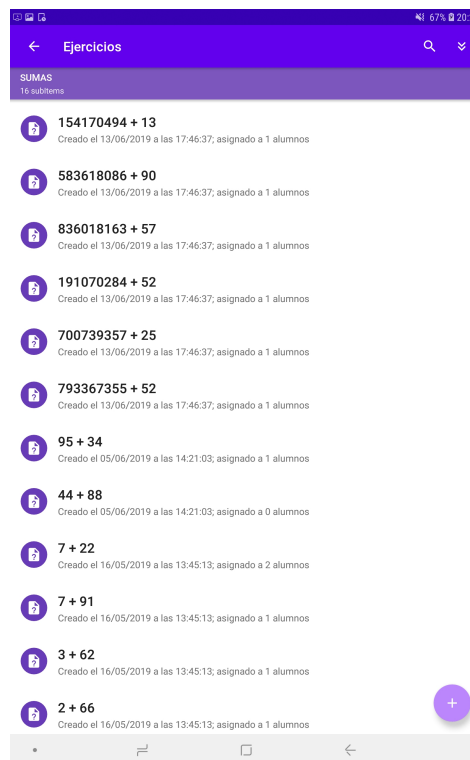
Figura B.1: Menú principal de MathematicAR

Ejercicios

Desde aquí podrás gestionar los ejercicios generados y realizar asignaciones a los estudiantes de la aplicación. La lista de ejercicios mostrada será la figura B.2(a) si dicha lista está vacía; y la lista de los ejercicios como se muestra en la figura B.2(b) en caso contrario.



(a) Lista de ejercicios vacía



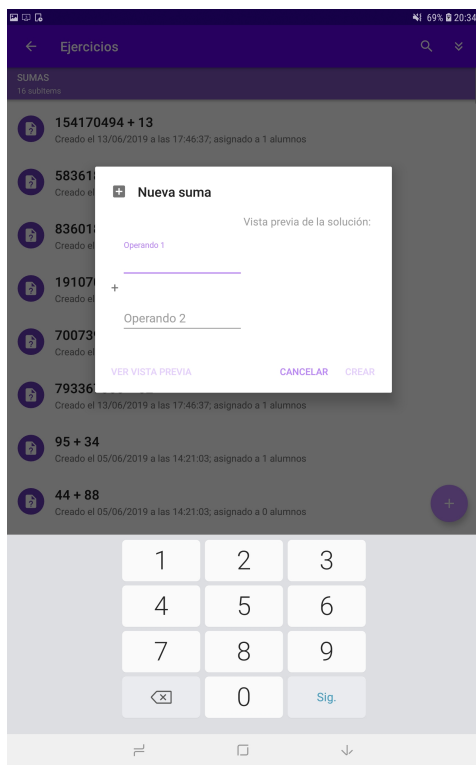
(b) Lista de ejercicios con elementos

Figura B.2: Lista con los ejercicios existentes

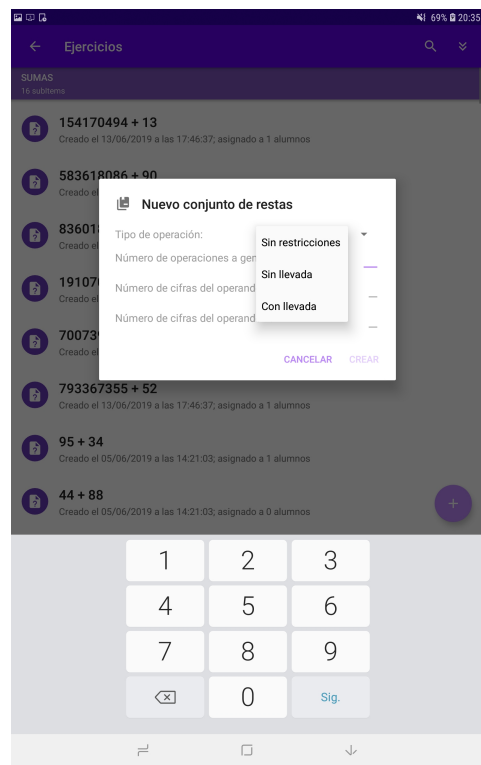
Con el botón de la parte inferior derecha se puede llevar a cabo la generación de operaciones aritméticas. Existen dos opciones: generar una operación o automatizar el proceso en función del tipo de operación. Independientemente de la selección, configura la operación como desees y confirma la creación de este. Las figuras B.3(a) y B.3(b) muestran los campos que se requieren en cada uno de los casos respectivamente. Un ejercicio está determinado de manera única por su primer y segundo operando y su operación; por lo que cuidado al crear ejercicios, ya que si existe no se creará.

Seleccionando uno o varios ejercicios, se pueden realizar las acciones indicadas en la figura B.4:

- Seleccionar todos los elementos.
- Imprimir las fichas de ejercicios de los alumnos que tienen asignado los seleccionados.
- Asignar los seleccionados a alumnos; abriendo el menú de los alumnos y permitiendo seleccionar aquellos a los que se asignárselos. Esta opción sólo está disponible cuando el elemento seleccionado es un ejercicio.
- Eliminar los seleccionados, que también eliminará las asignaciones que contengan dichos ejercicios.



(a) Operación aritmética simple



(b) Conjunto de operaciones aritméticas

Figura B.3: Posibles maneras de generar operaciones

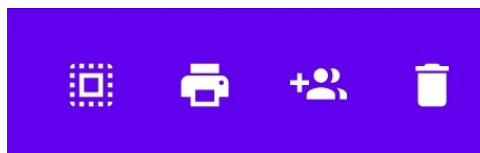
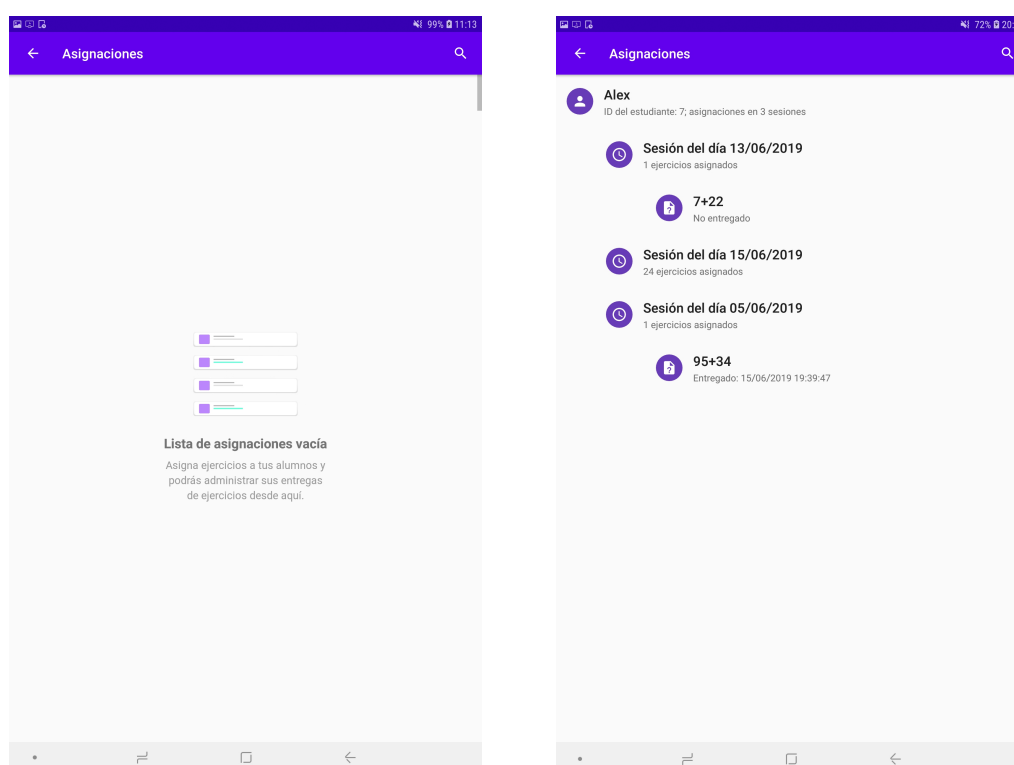


Figura B.4: Opciones disponibles sobre uno o varios ejercicios seleccionados

Asignaciones

En Asignaciones se permite administrar las entregas de ejercicios de los alumnos. La vista mostrada será la de la figura B.5(a) si dicha lista está vacía; y la lista de las asignaciones como se muestra en la figura B.5(b) en caso contrario.

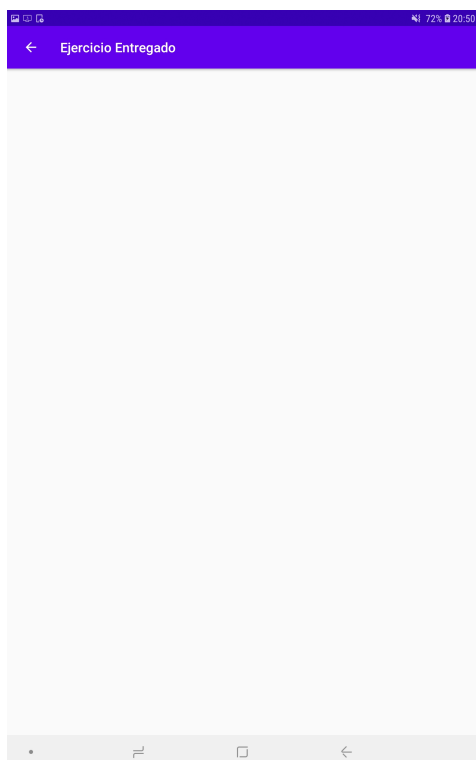


(a) Lista de asignaciones vacía

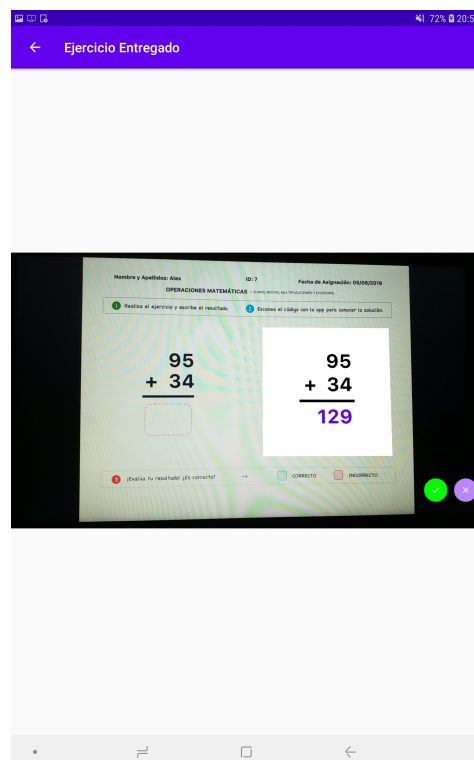
(b) Lista de asignaciones con elementos

Figura B.5: Vista de las asignaciones

Pulsando sobre el ejercicio del que se desea ver la entrega efectuada, se mostrará dicha entrega en caso de que se haya realizado por parte del alumno, o una pantalla en blanco en caso contrario; de acuerdo con lo observado en la figuras B.6(b) y B.6(a), respectivamente.



(a) Entrega no realizada

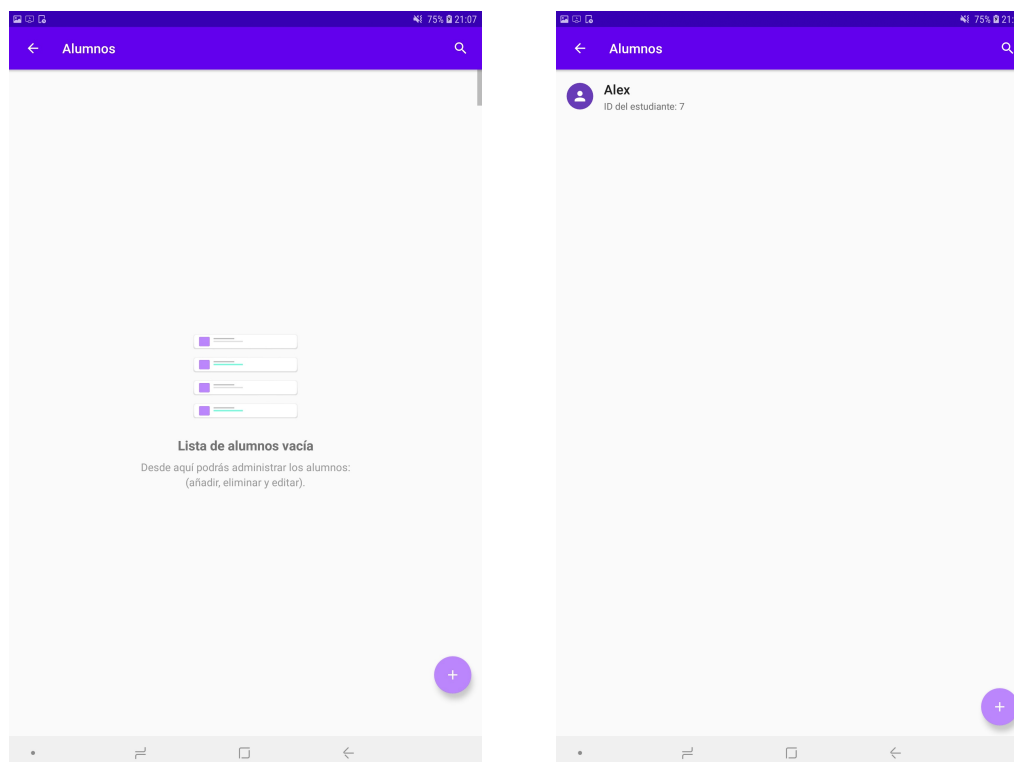


(b) Entrega realizada

Figura B.6: Posibles visiones de una entrega concreta

Alumnos

Desde Alumnos se podrá gestionar el conjunto de los alumnos a los que asignar ejercicios. La lista de alumnos mostrada será la figura B.7(a) si la lista está vacía; y la lista de los alumnos como se muestra en la figura B.7(b) en caso contrario.



(a) Lista de alumnos vacía

(b) Lista de alumnos con elementos

Figura B.7: Lista con los alumnos existentes

Con el botón de la parte inferior derecha se puede llevar a cabo la creación de un nuevo alumno. La figura B.8 muestra el diálogo con los campos que se requieren para llevar a cabo esta creación.

Nuevo alumno

Nombre

Apellidos

CANCELAR CREAM

Figura B.8: Diálogo mostrado al crear un nuevo alumno

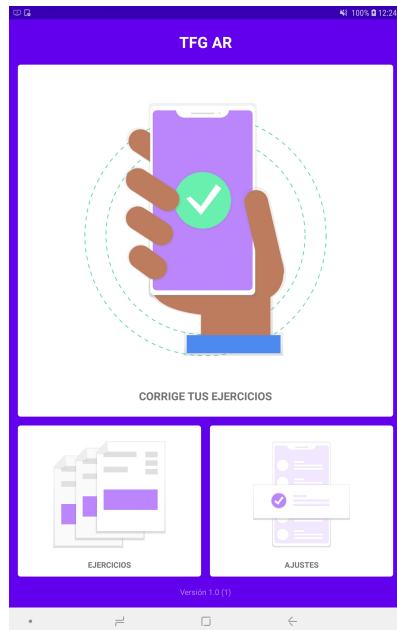
Seleccionando uno o varios alumnos, se pueden realizar las acciones indicadas en la figura B.9:

- Seleccionar todos los alumnos.
- Editar los datos de un alumno: esta acción está disponible sólo si un alumno ha sido seleccionado.
- Eliminar los alumnos seleccionados, que también eliminará las asignaciones que contengan dichos alumnos.

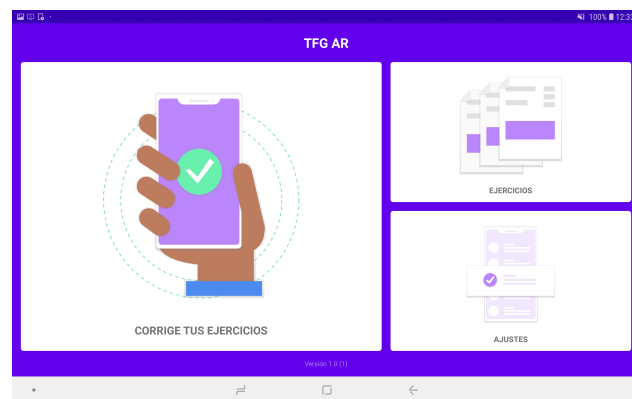


Figura B.9: Opciones disponibles sobre uno o varios alumnos seleccionados

CAPTURAS DEL ESTADO INICIAL



(a) Orientación vertical



(b) Orientación horizontal

Figura C.1: Menú principal de la aplicación

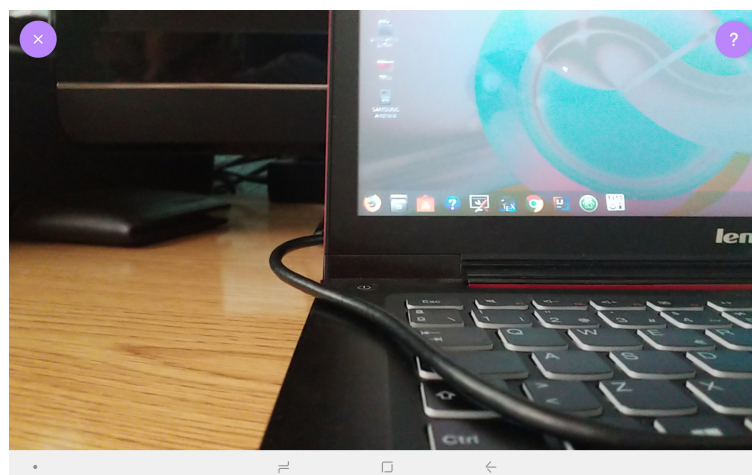


Figura C.2: Vista inicial de Corrige tus Ejercicios

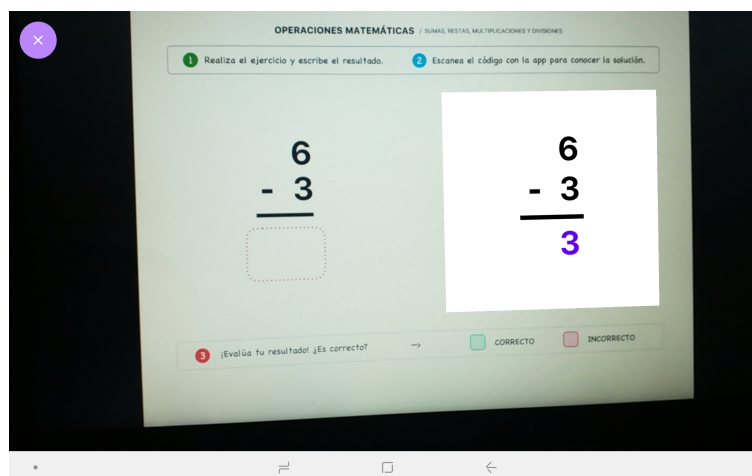
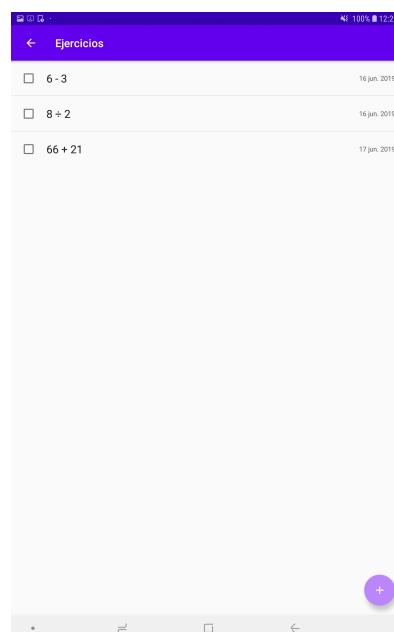


Figura C.3: Resultado de la operación de la ficha de ejercicios



(a) Lista de ejercicios vacía



(b) Lista de ejercicios con elementos

Figura C.4: Lista con los ejercicios existentes

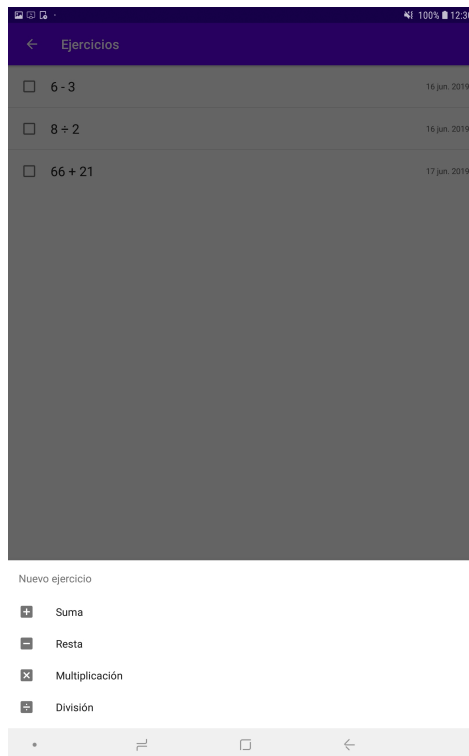


Figura C.5: Menú con el tipo de operación a crear

Nueva multiplicación

1er operando × 2do operando

[CANCELAR](#) [CREAR](#)

Figura C.6: Diálogo para la creación de la operación

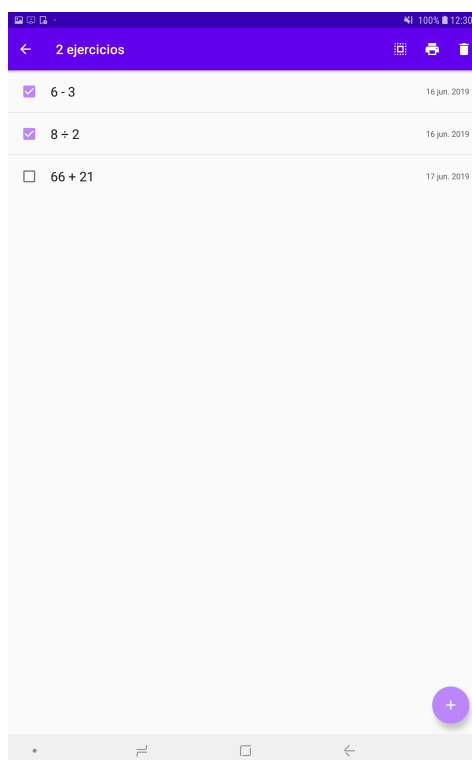


Figura C.7: Selección de ejercicios

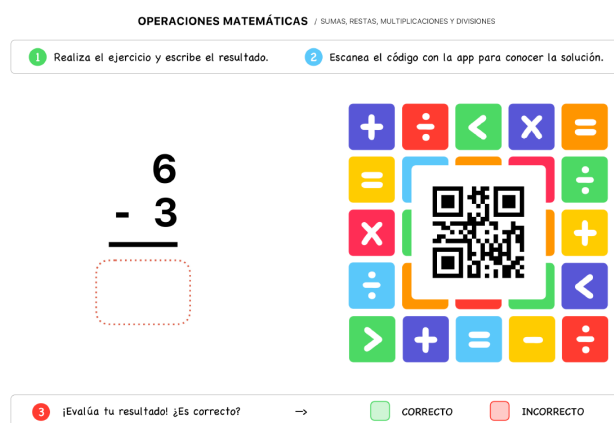


Figura C.8: Ejemplo de ficha de ejercicios

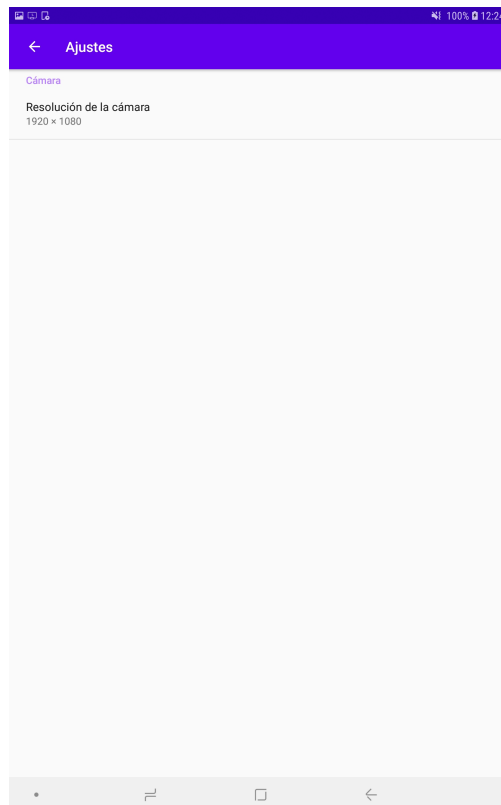


Figura C.9: Vista de los ajustes disponibles

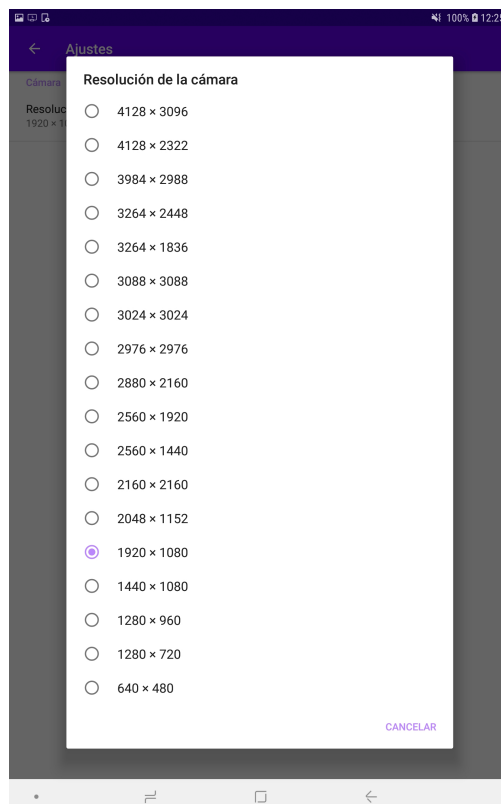
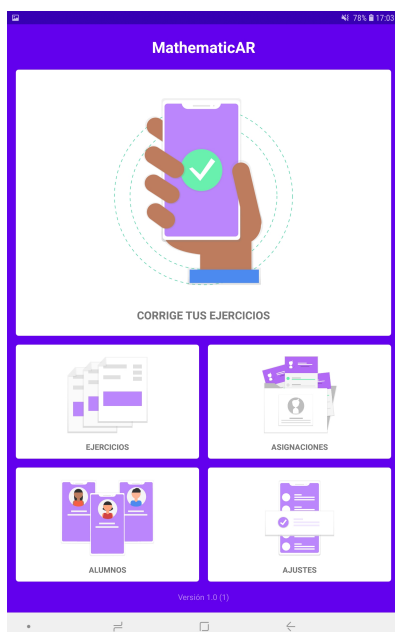


Figura C.10: Diálogo con las resoluciones de cámara disponibles

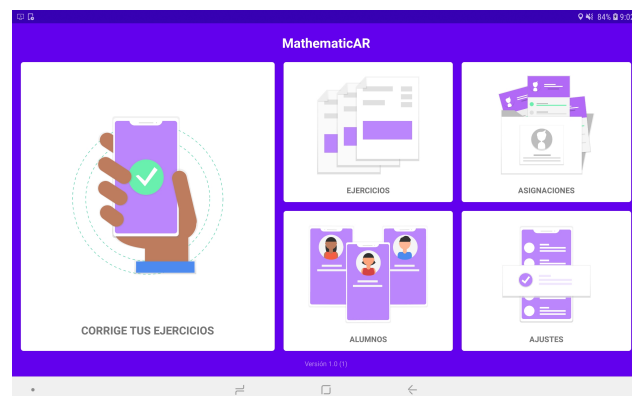
CAPTURAS DEL ESTADO FINAL



Figura D.1: Vista inicial previa a la carga del menú principal



(a) Orientación vertical



(b) Orientación horizontal

Figura D.2: Menú principal de MathematicAR

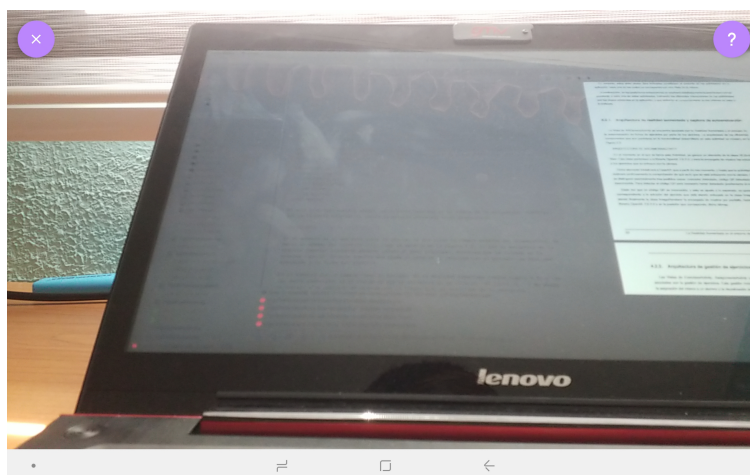


Figura D.3: Vista inicial de Corrige tus Ejercicios

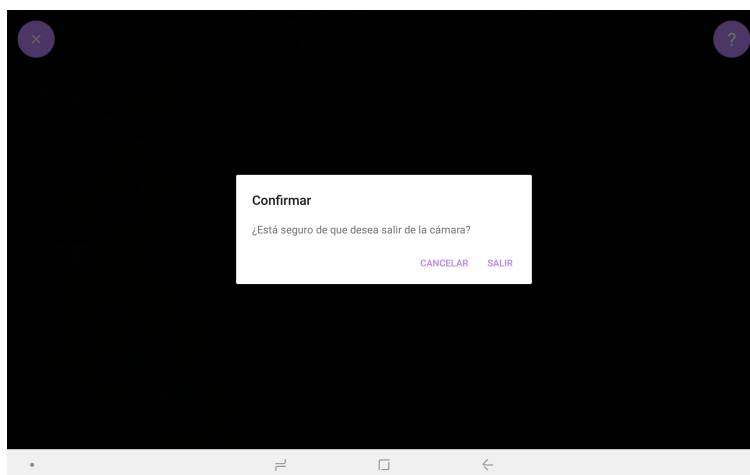


Figura D.4: Diálogo relativo a salir al menú principal

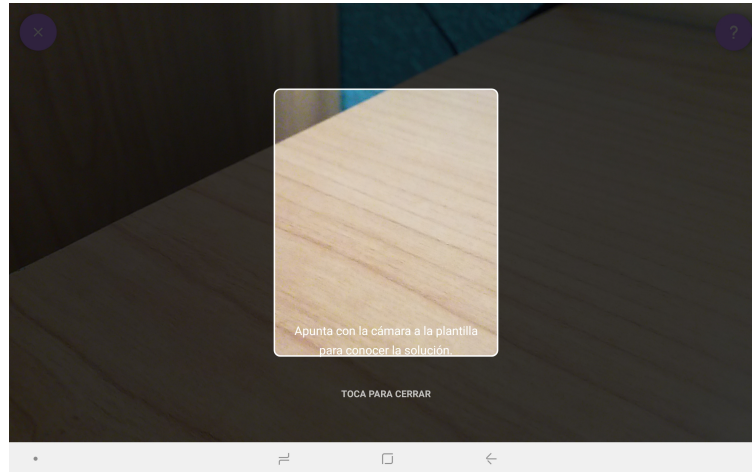
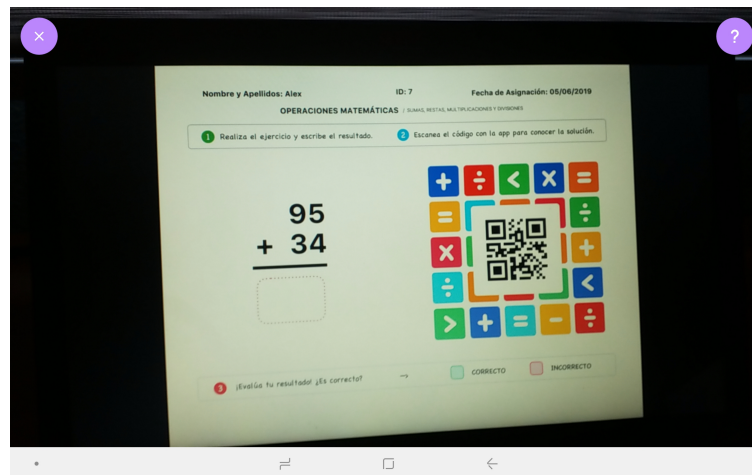
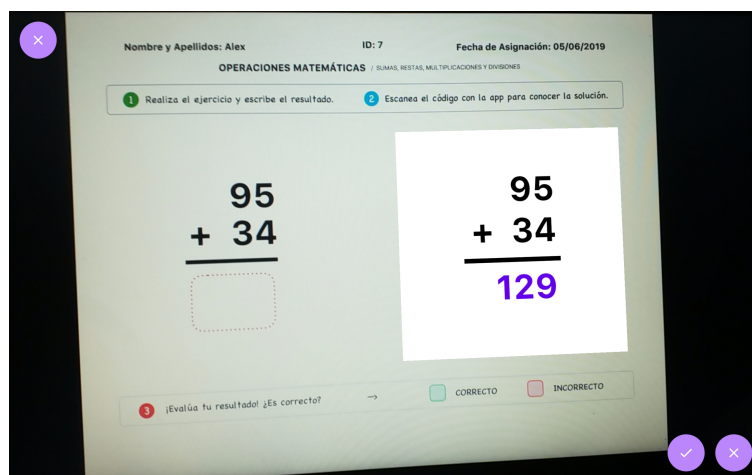


Figura D.5: Mensaje de ayuda



(a) Marcador y QR no detectados



(b) Marcador y QR detectados

Figura D.6: Transición cuando la cámara capta el marcador y lee el código QR

Entrega realizada

El ejercicio se ha entregado correctamente.

ACEPTAR

(a) Entrega correcta

Error en la entrega

El ejercicio que se está intentando realizar ya se ha entregado.

ACEPTAR

(b) Entrega ya realizada

Error en el ejercicio

El ejercicio no se corresponde con este dispositivo.

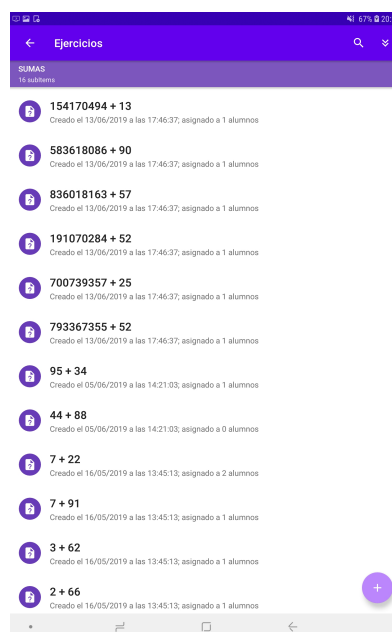
ACEPTAR

(c) Entrega no existente en el dispositivo

Figura D.7: Posibles mensajes mostrados al realizar una entrega

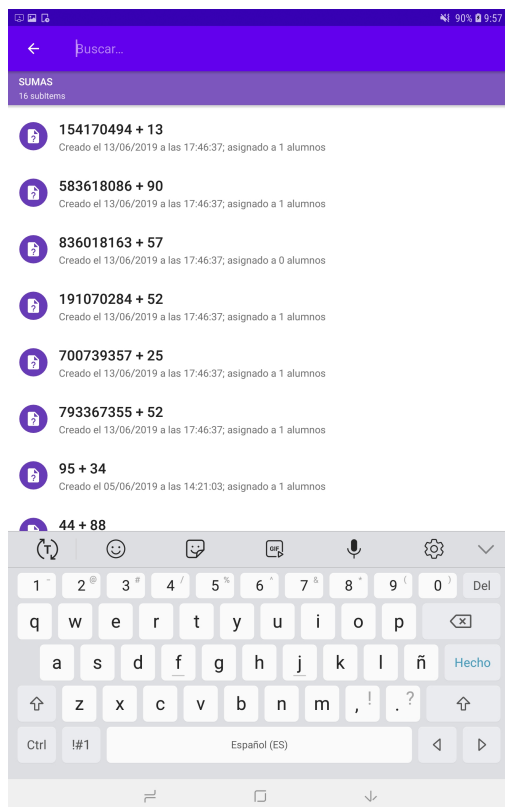


(a) Lista de ejercicios vacía

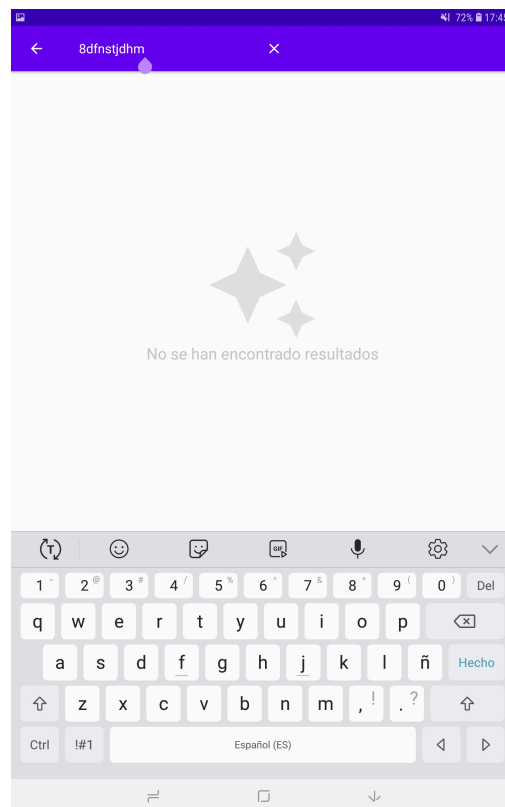


(b) Lista de ejercicios con elementos

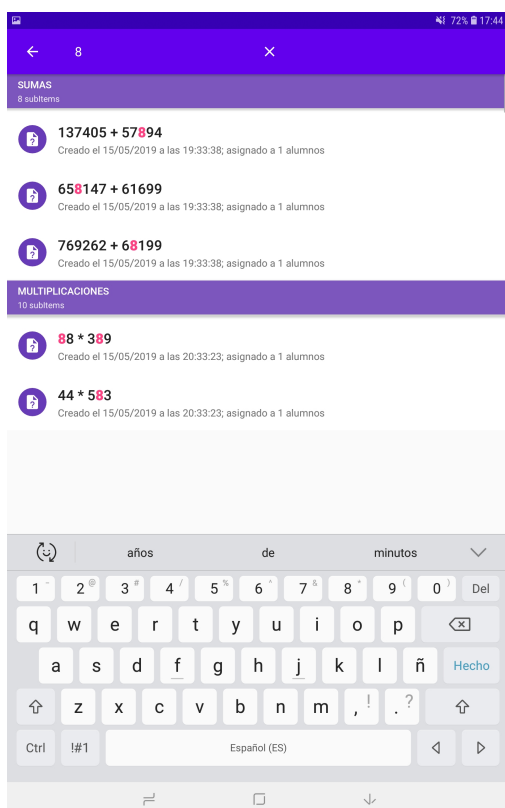
Figura D.8: Lista con los ejercicios existentes



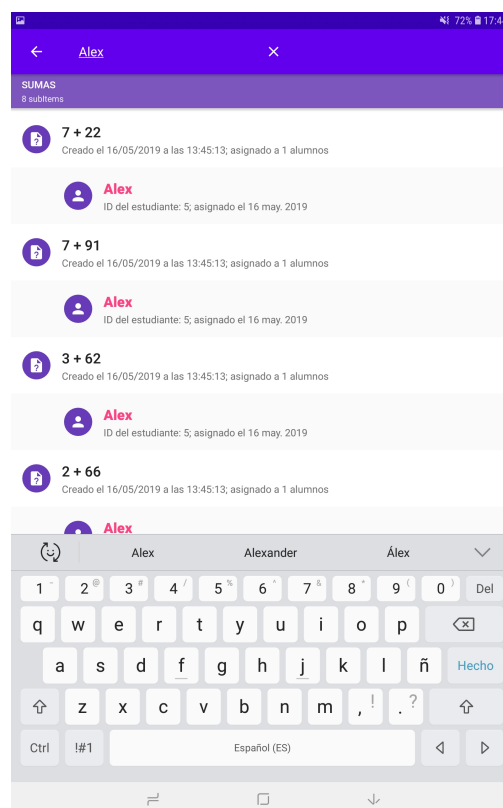
(a) Estado inicial de una búsqueda



(b) Sin resultados de búsqueda



(c) Resultados de búsqueda de ejercicios



(d) Resultados de búsqueda de alumnos

Figura D.9: Posibles estados de la opción de búsquedas

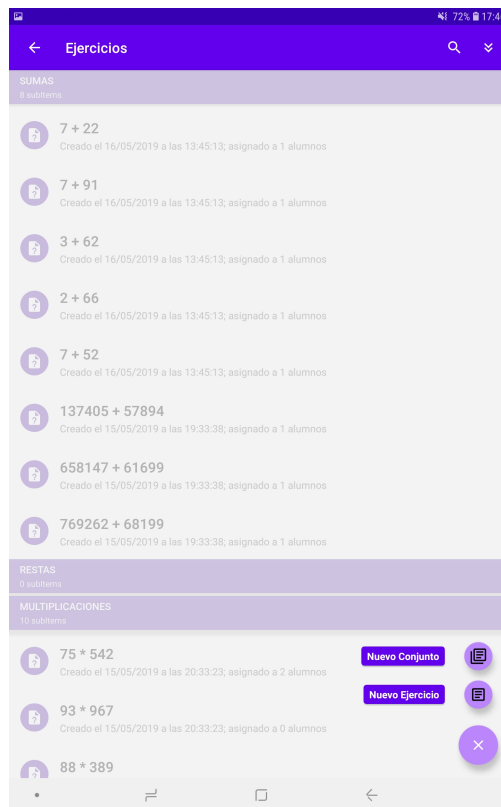
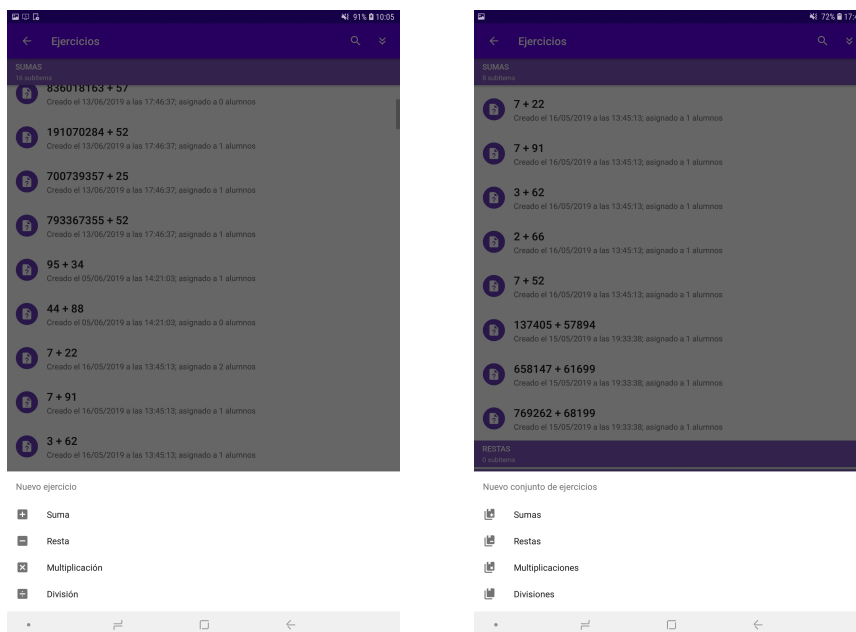


Figura D.10: Acciones disponibles para la creación de ejercicios



(a) Operación Simple

(b) Conjunto de operaciones

Figura D.11: Diálogo del tipo de operación a crear

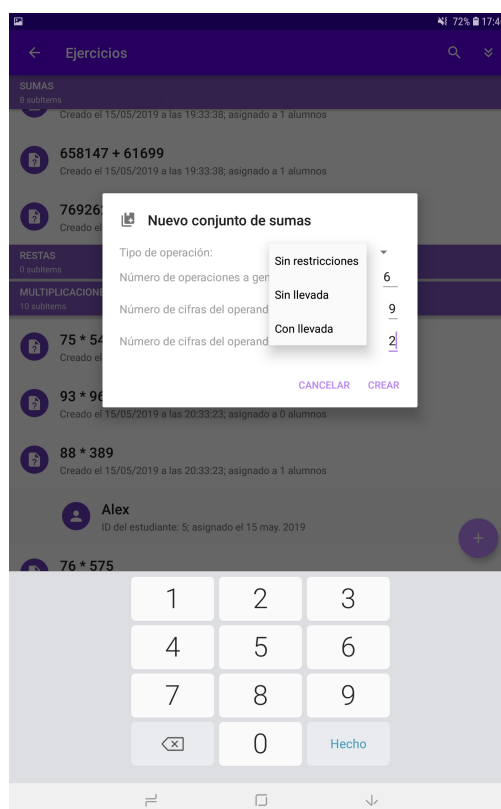
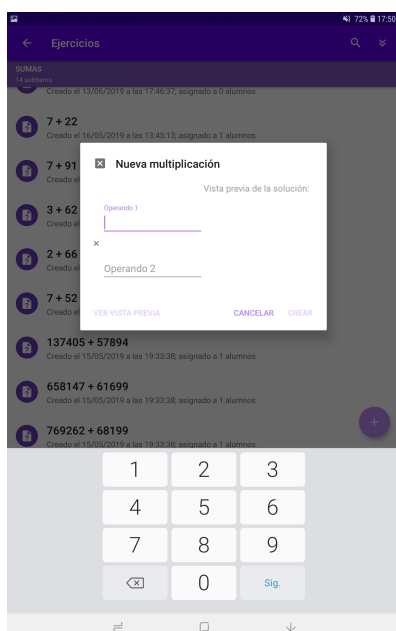
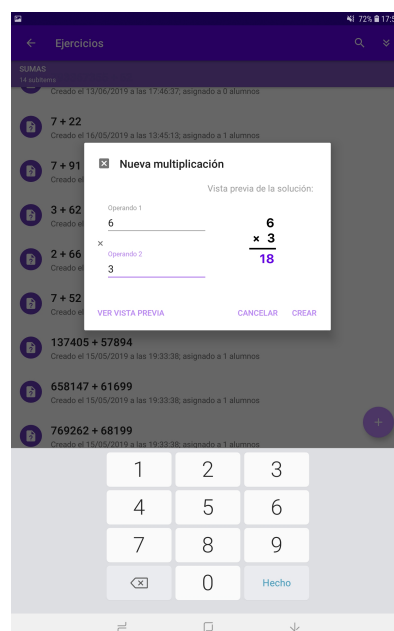


Figura D.12: Diálogo para la creación de un conjunto de operaciones

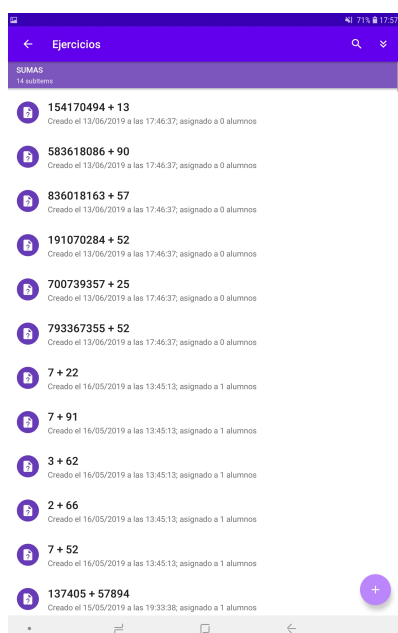


(a) Diálogo inicial

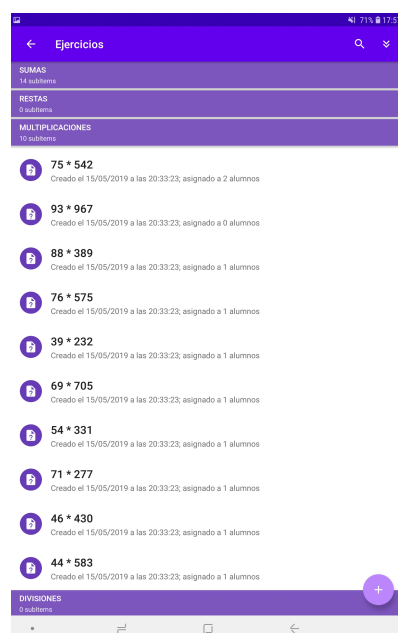


(b) Vista previa

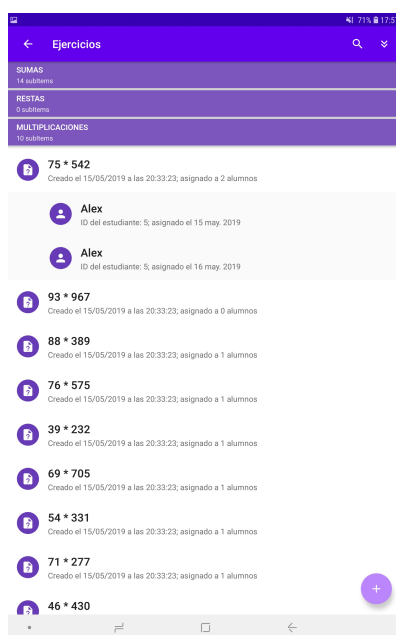
Figura D.13: Diálogo para la creación de una operación



(a) Estado inicial

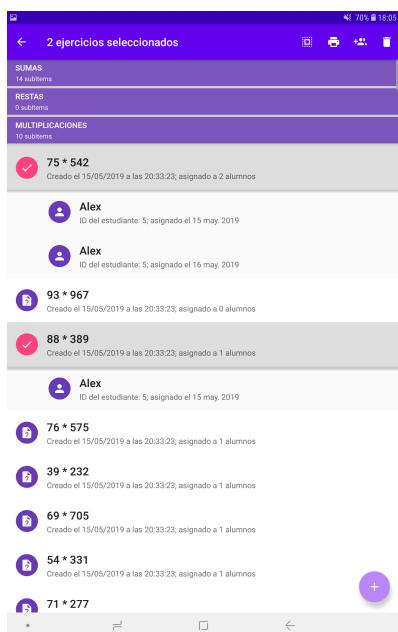


(b) Colapso de tipos de operación

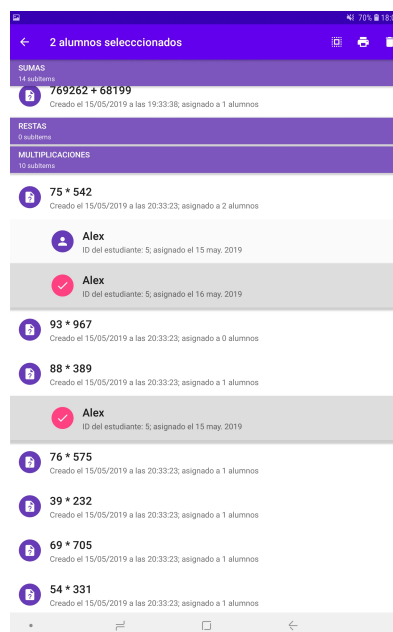


(c) Expansión de ejercicio

Figura D.14: Proceso de colapso y expansión de los elementos de la interfaz

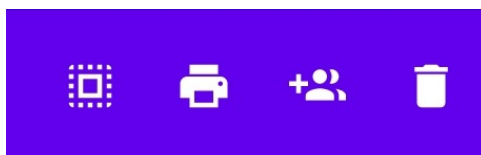


(a) Selección de ejercicios

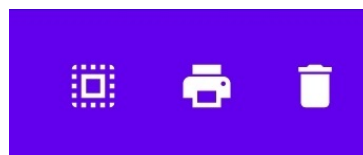


(b) Selección de alumnos

Figura D.15: Selección de elementos de la interfaz

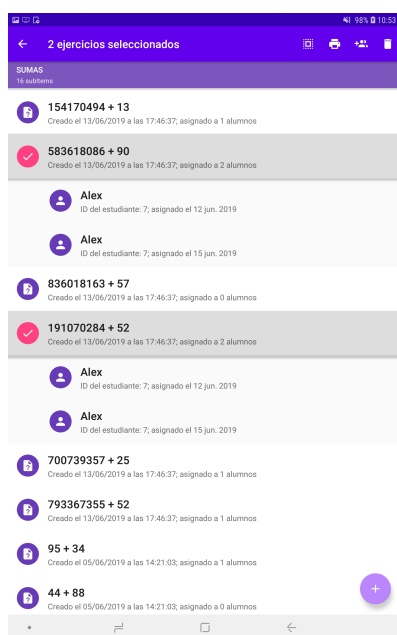


(a) Opciones si se seleccionan ejercicios

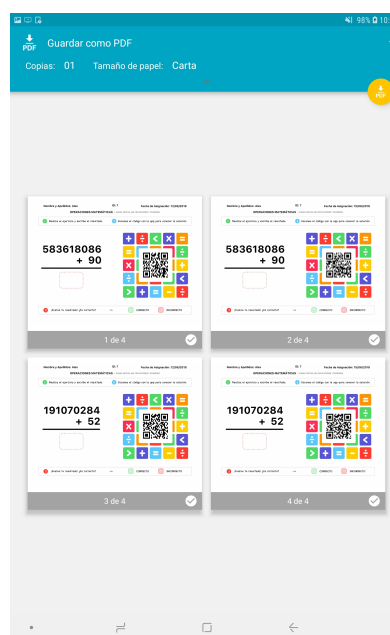


(b) Opciones si se seleccionan alumnos

Figura D.16: Opciones disponibles sobre uno o varios elementos seleccionados

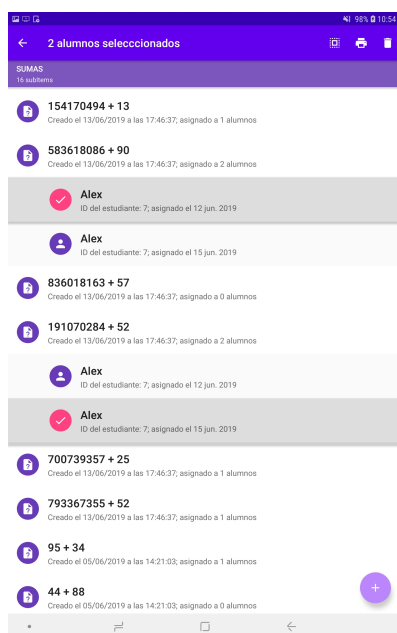


(a)

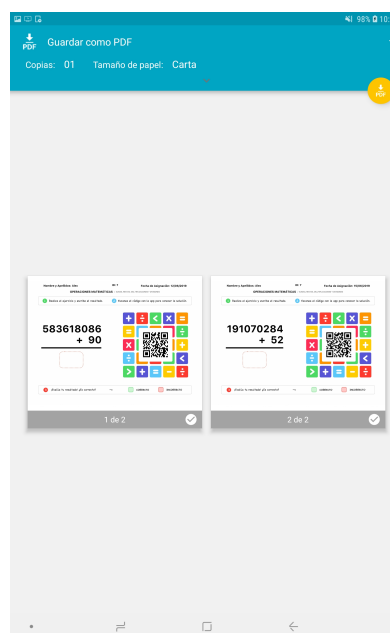


(b)

Figura D.17: Impresión de un ejercicio seleccionado

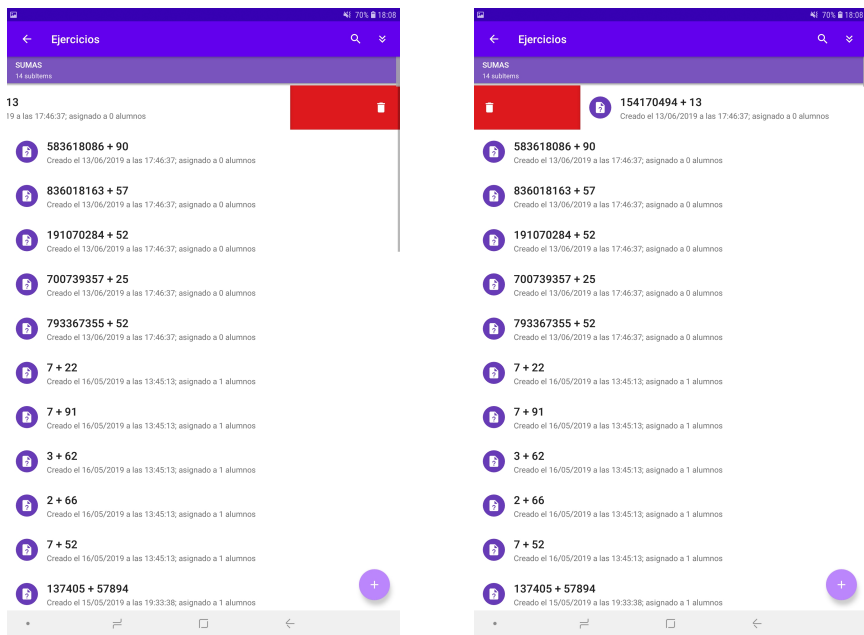


(a)



(b)

Figura D.18: Impresión de un ejercicio de un alumno seleccionado



(a) Acción de *swipe* a la izquierda

(b) Acción de *swipe* a la derecha

Figura D.19: Proceso de borrado mediante *swipe*

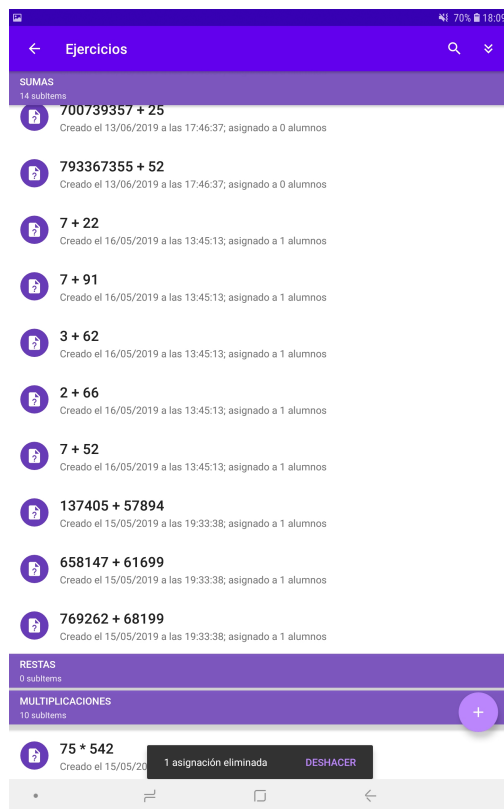
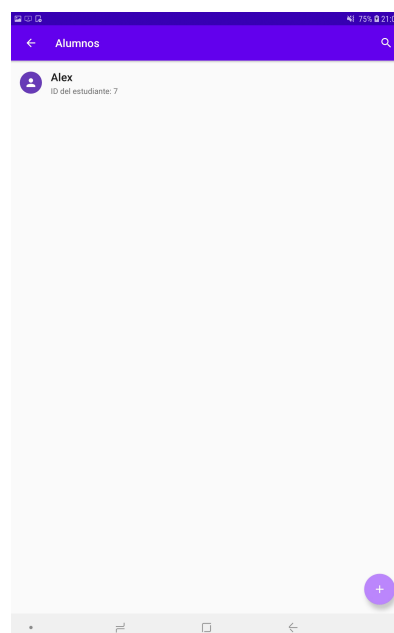


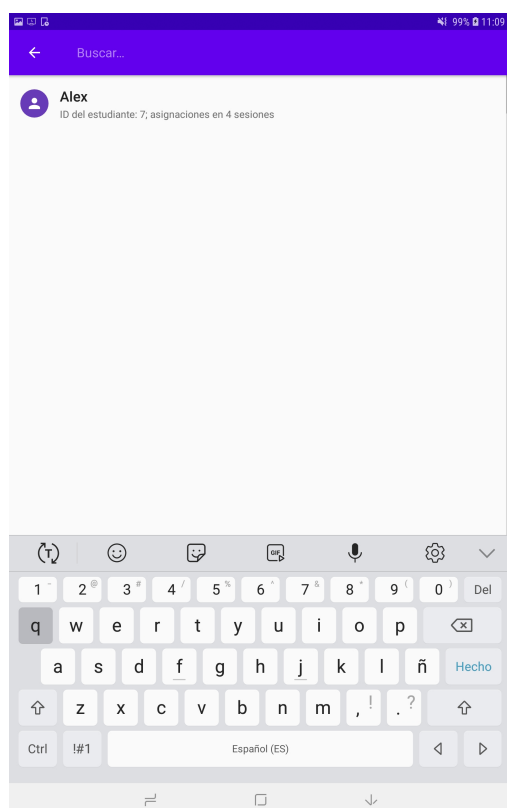
Figura D.20: Mensaje de eliminación de un elemento



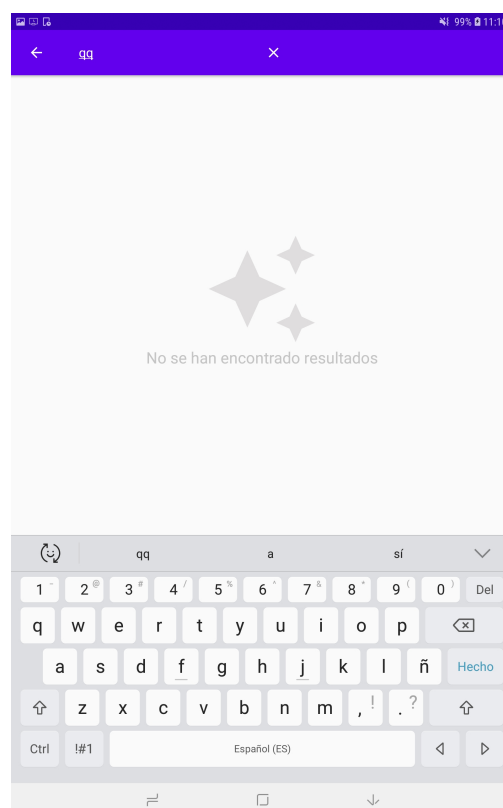
(a) Lista de asignaciones vacía



(b) Lista de asignaciones con elementos

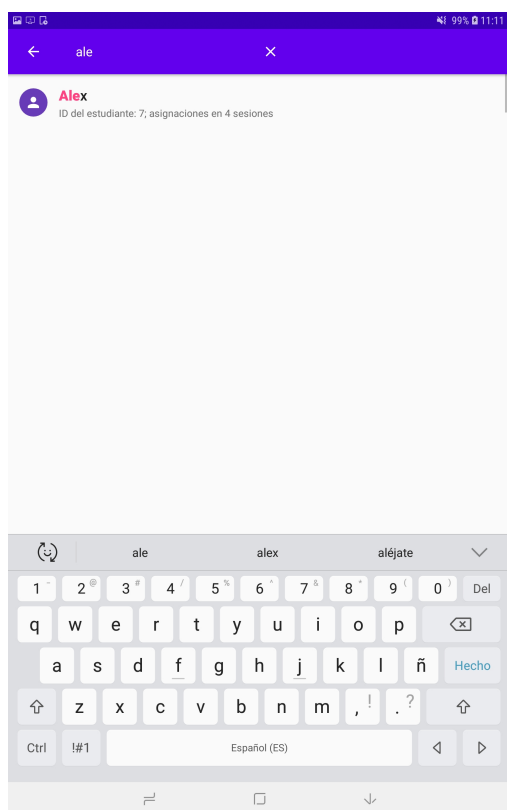
Figura D.21: Lista con las asignaciones existentes

(a) Estado inicial de una búsqueda

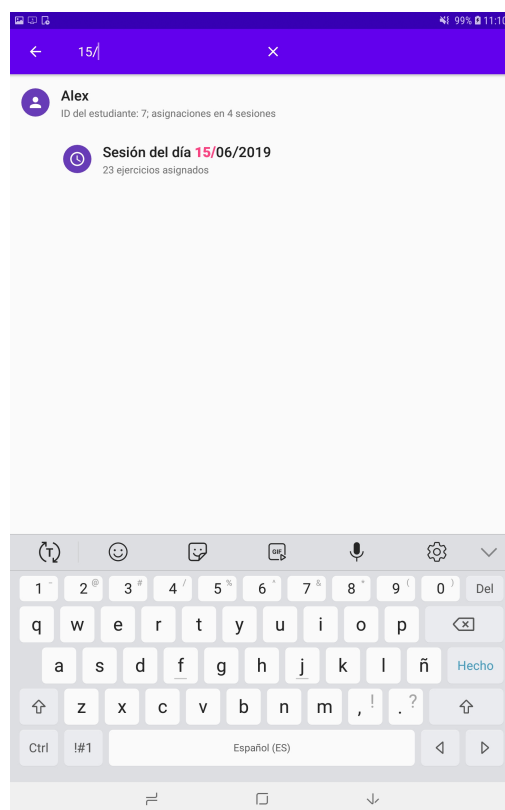


(b) Sin resultados de búsqueda

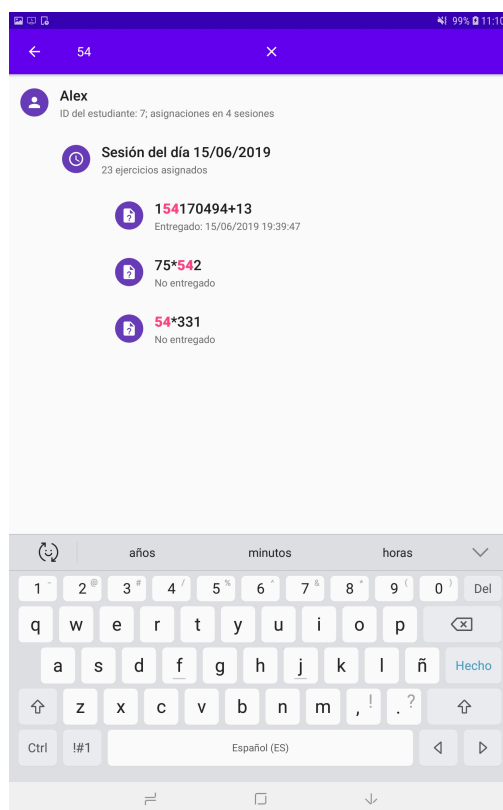
Figura D.22: Posibles estados de la opción de búsquedas



(a) Resultados de búsqueda de alumnos



(b) Resultados de búsqueda de sesión



(c) Resultados de búsqueda de ejercicios

Figura D.23: Posibles estados de la opción de búsquedas exitosas

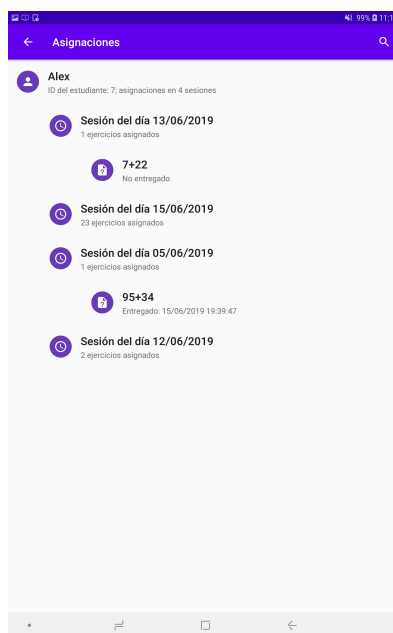
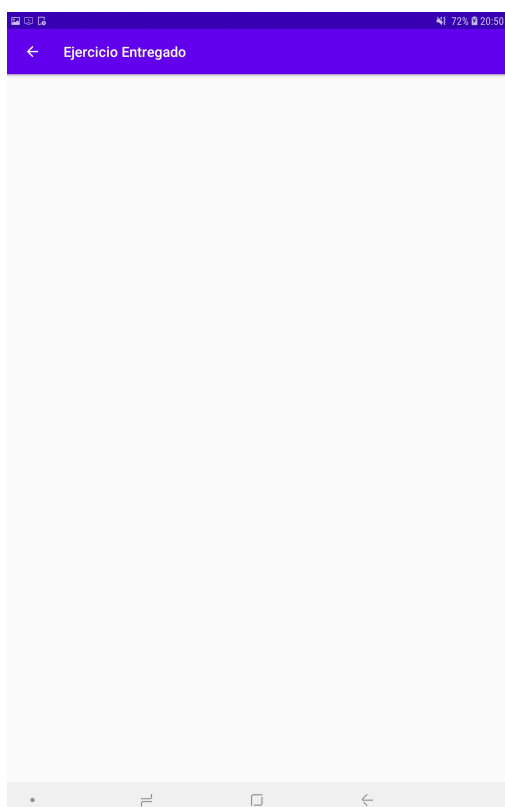
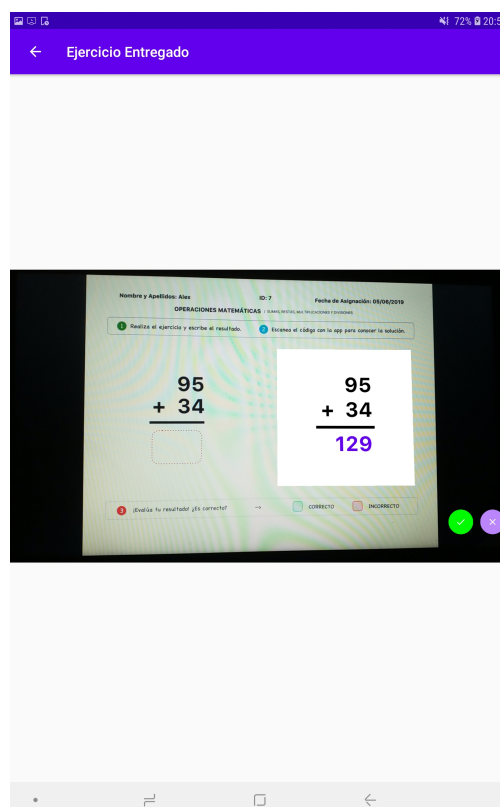


Figura D.24: Interfaz tras expandir elementos



(a) Entrega no realizada

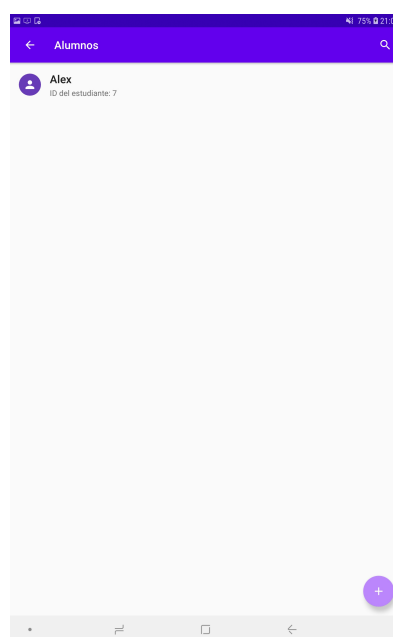


(b) Entrega realizada

Figura D.25: Posibles visiones de una entrega concreta

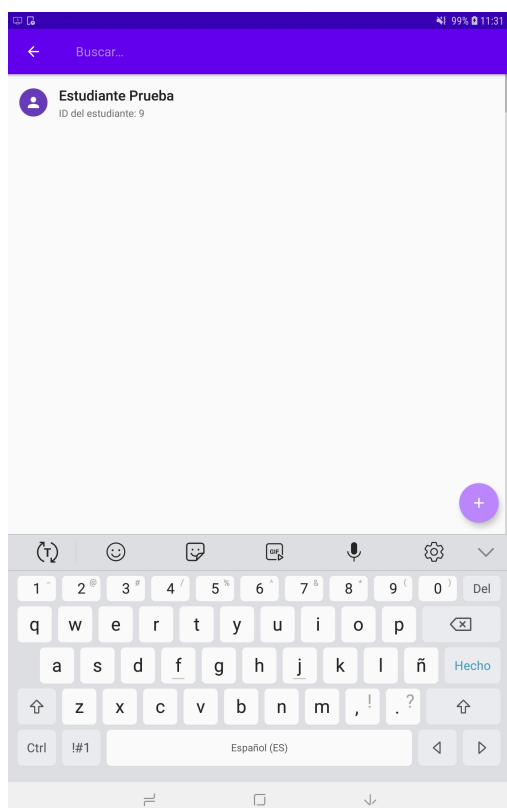


(a) Lista de alumnos vacía

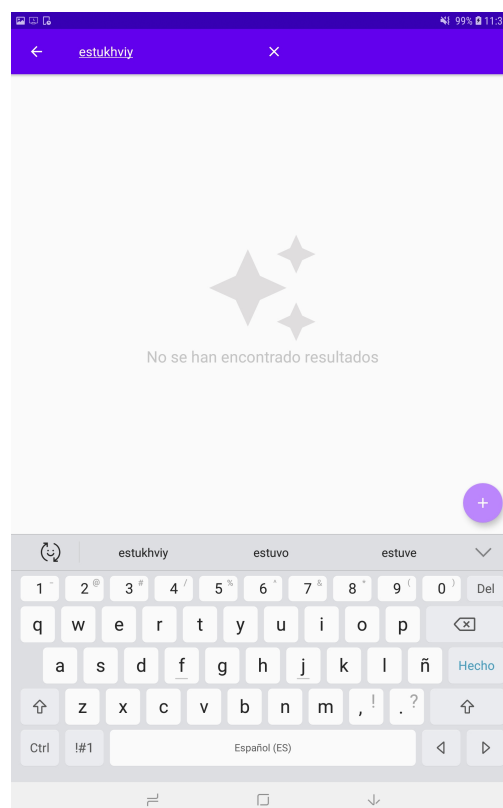


(b) Lista de alumnos con elementos

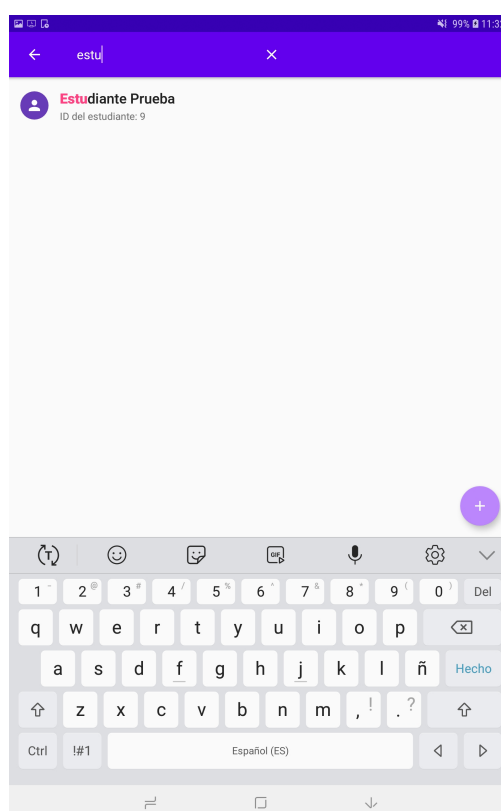
Figura D.26: Lista con los alumnos existentes



(a) Estado inicial de una búsqueda



(b) Sin resultados de búsqueda



(c) Resultados de búsqueda de alumnos

Figura D.27: Posibles estados de la opción de búsquedas

Nuevo alumno

Nombre

Apellidos

Figura D.28: Diálogo para la creación de alumnos

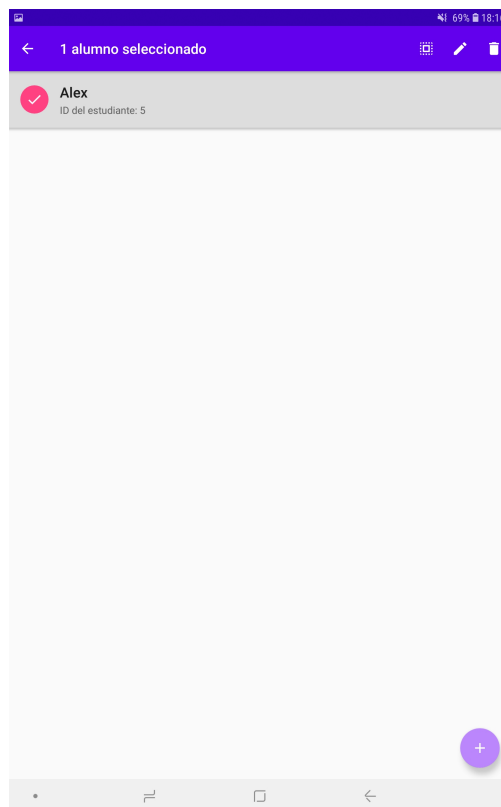


Figura D.29: Selección de alumnos

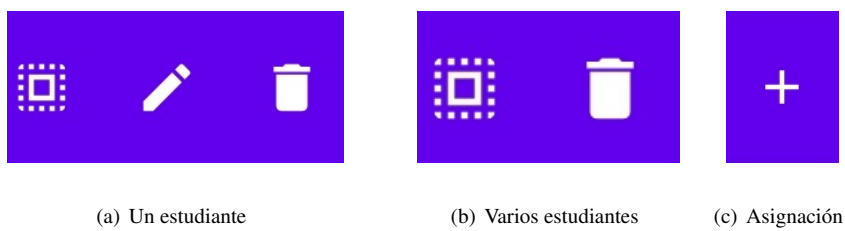


Figura D.30: Opciones disponibles sobre uno o varios alumnos seleccionados

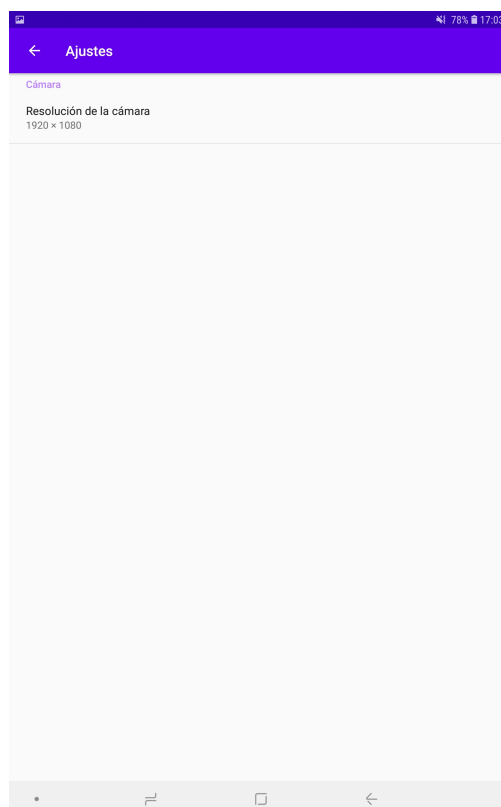


Figura D.31: Vista de los ajustes disponibles

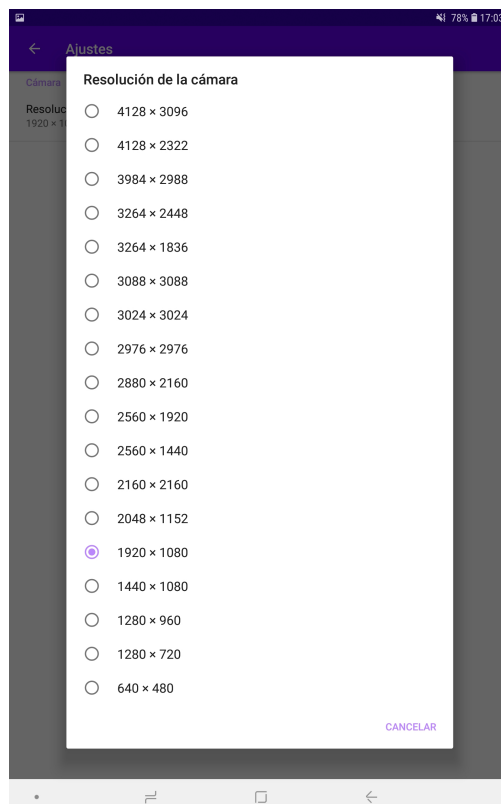


Figura D.32: Diálogo con las resoluciones de cámara disponibles

EJEMPLO DE FICHA DE EJERCICIOS

E

Nombre y Apellidos: Alex ID: 7 Fecha de Asignación: 05/06/2019

OPERACIONES MATEMÁTICAS / SUMAS, RESTAS, MULTIPLICACIONES Y DIVISIONES

- 1 Realiza el ejercicio y escribe el resultado.
- 2 Escanea el código con la app para conocer la solución.

$$\begin{array}{r} 95 \\ + 34 \\ \hline \end{array}$$



3 ¡Evalúa tu resultado! ¿Es correcto?



CORRECTO



INCORRECTO

Figura E.1: Ejemplo de ficha de ejercicios de MathematicAR

